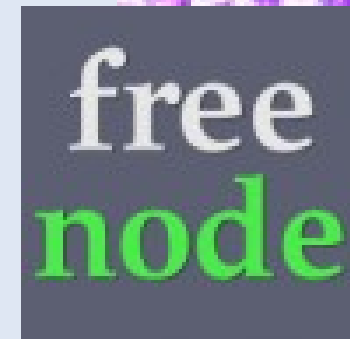
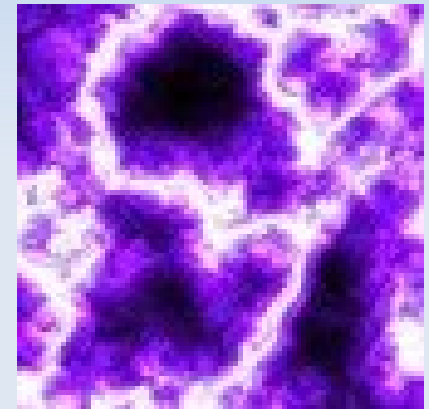


Introduction

References and Presentation at:
<http://www.elinux.org/elce-i2c>

Introduction

- Dave Anders aka prpplague



Introduction

- Dave Anders aka prpplague
- Currently Contracted with TI



Introduction

- Dave Anders aka prpplague
- Currently Contracted with TI
- Partners in TinCanTools



Introduction

- Dave Anders aka prpplague
- Currently Contracted with TI
- Partners in TinCanTools
- Board Bring: You, Me, and I2C

Introduction

- Dave Anders aka prpplague
- Currently Contracted with TI
- Partners in TinCanTools
- Board Bring: You, Me, and I2C
 - Communication Principles

Introduction

- Dave Anders aka prpplague
- Currently Contracted with TI
- Partners in TinCanTools
- Board Bring: You, Me, and I2C
 - Communication Principles
 - Drivers and Software Tools

Introduction

- Dave Anders aka prpplague
- Currently Contracted with TI
- Partners in TinCanTools
- Board Bring: You, Me, and I2C
 - Communication Principles
 - Drivers and Software Tools
 - Board Bringup Use Cases

Communication Principles

- Asynchronous Communication

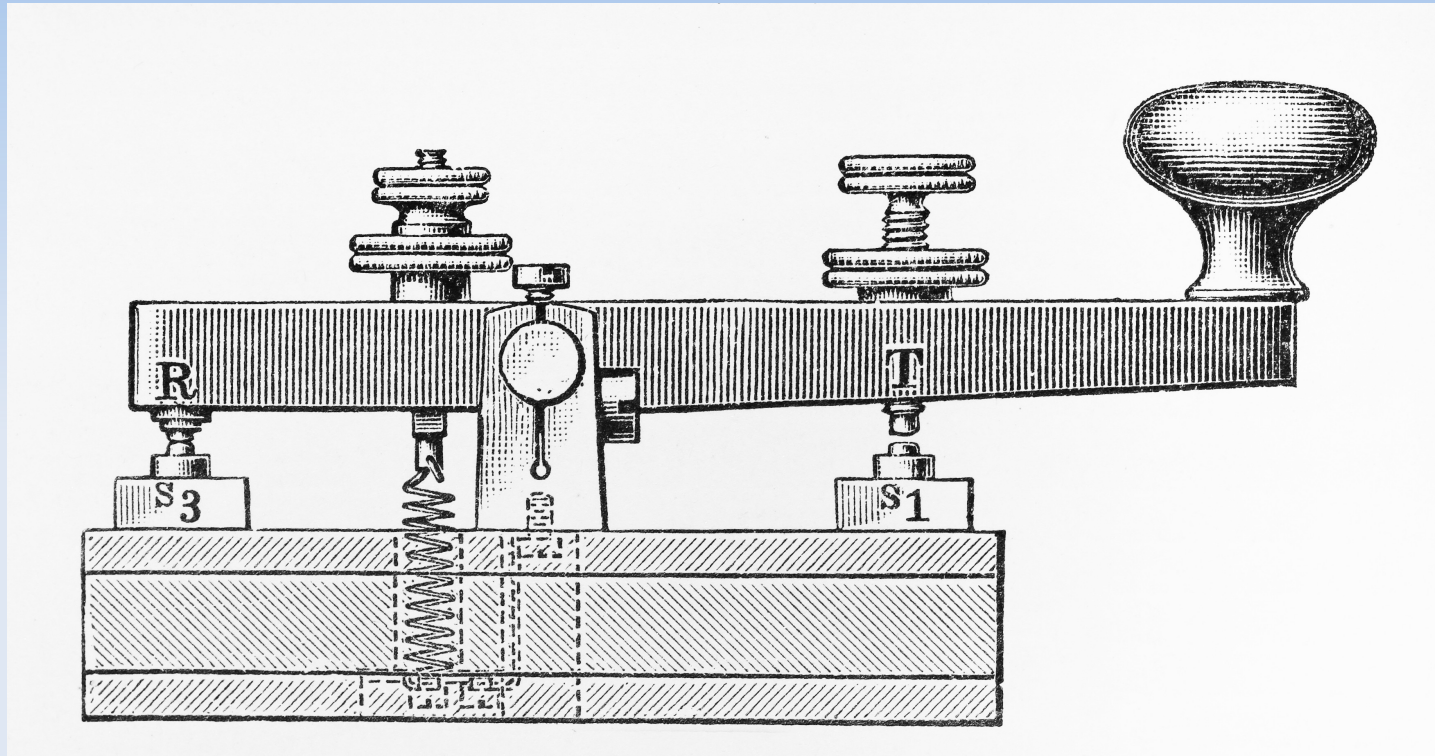
Communication Principles

- Asynchronous Communication
 - No External Clock Signal

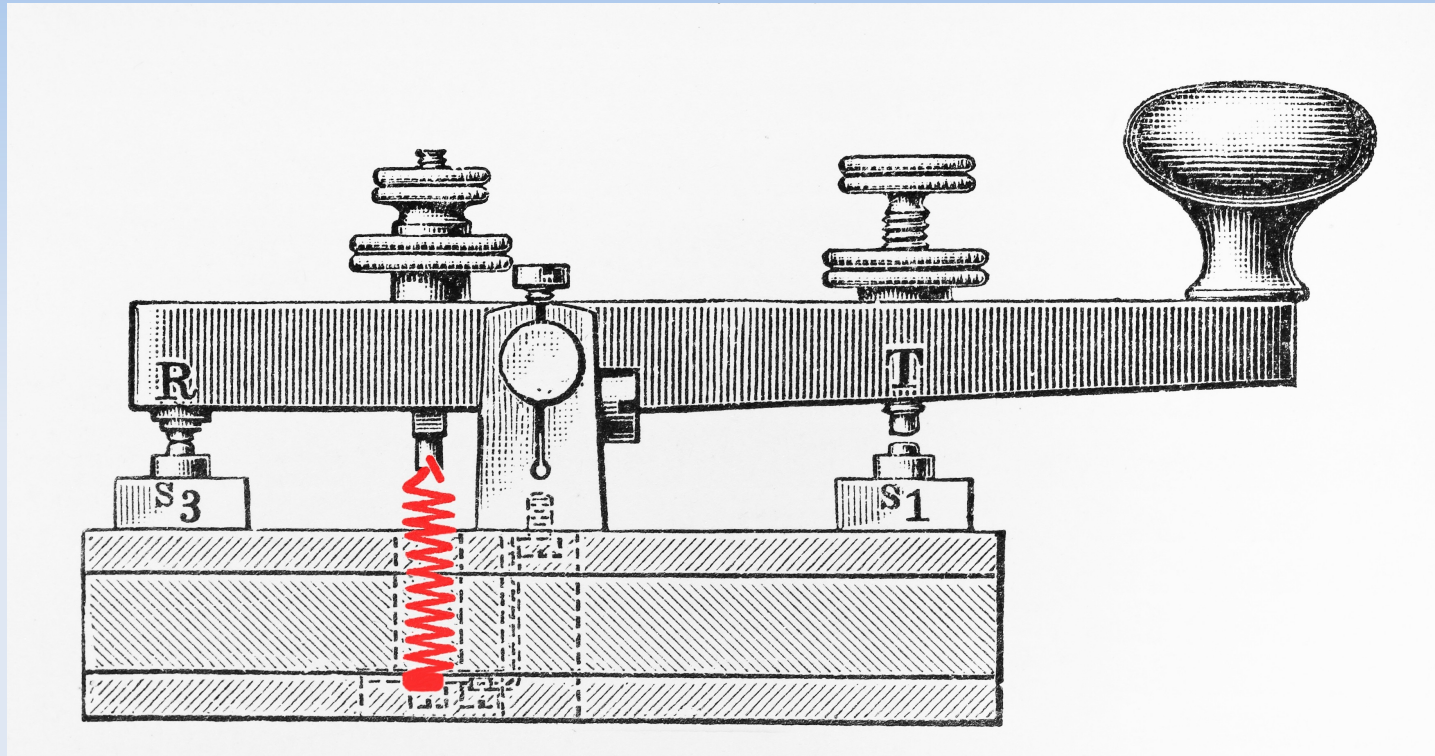
Communication Principles

- Asynchronous Communication
 - No External Clock Signal
 - Morse Code / Telegraph

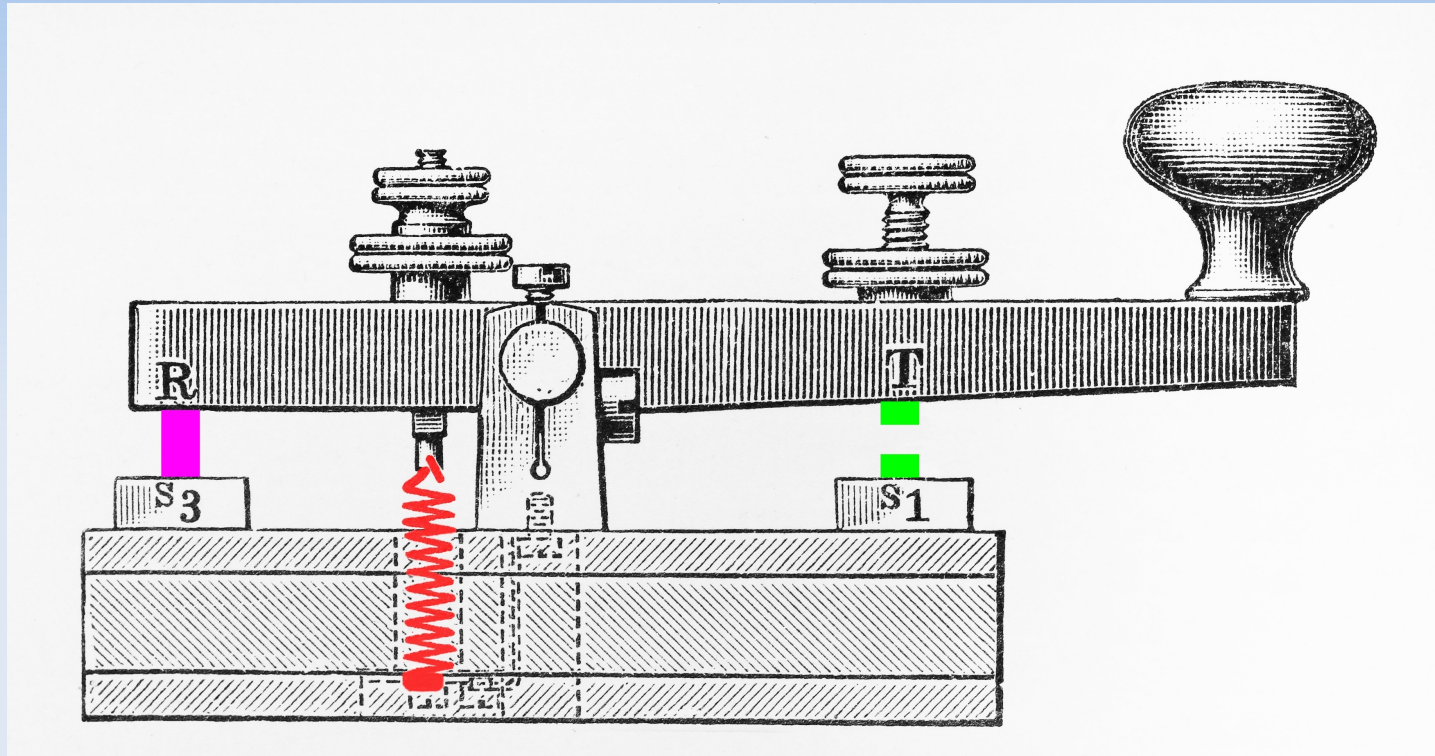
Communication Principles



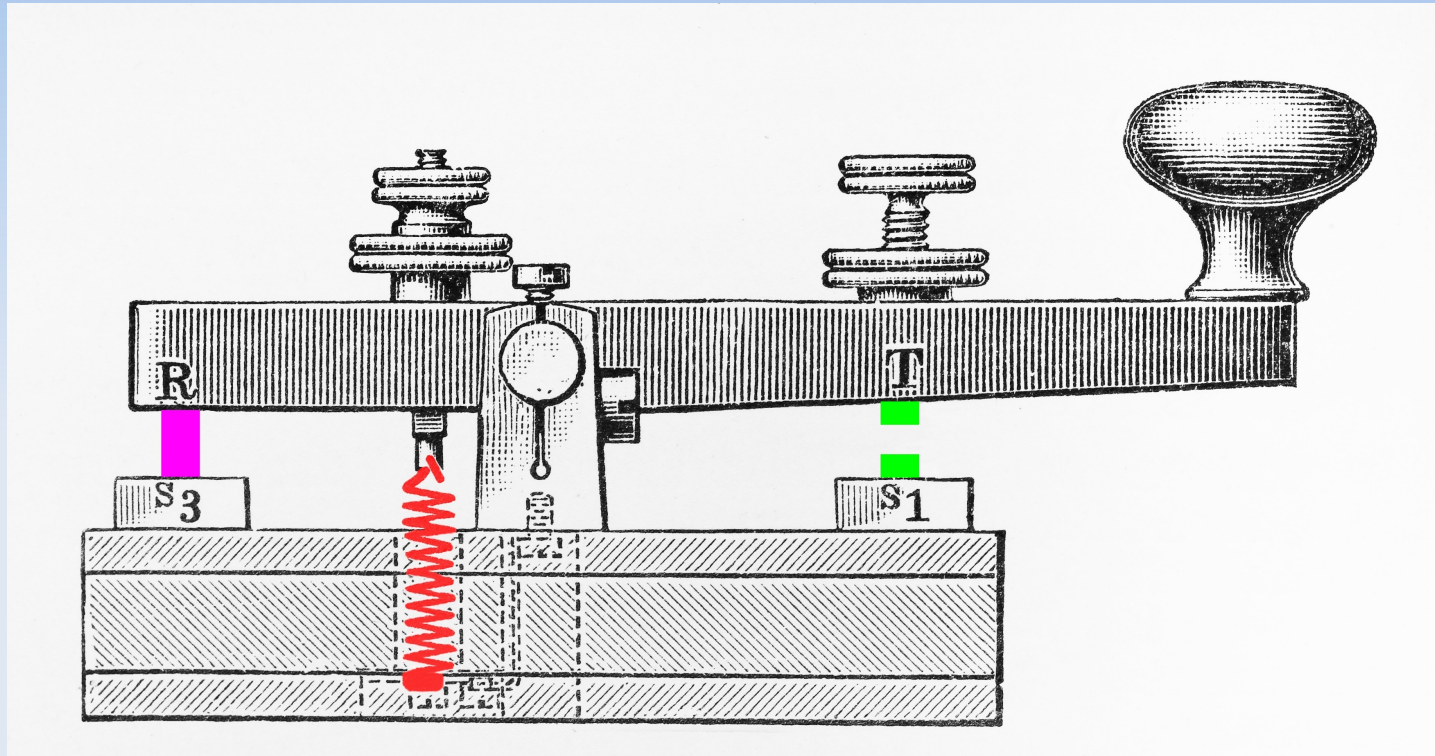
Communication Principles



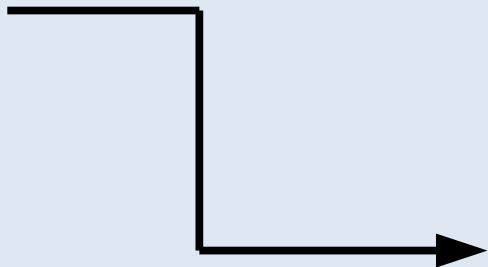
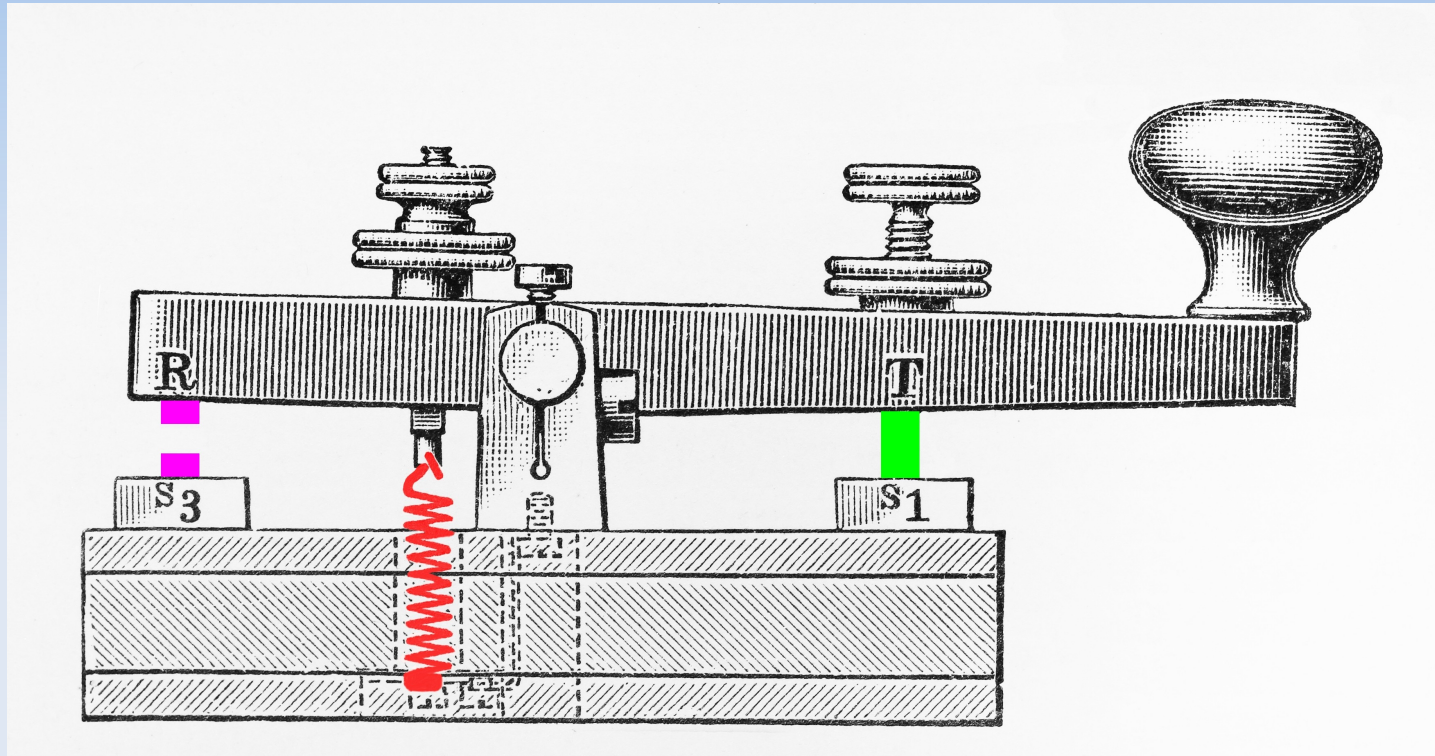
Communication Principles



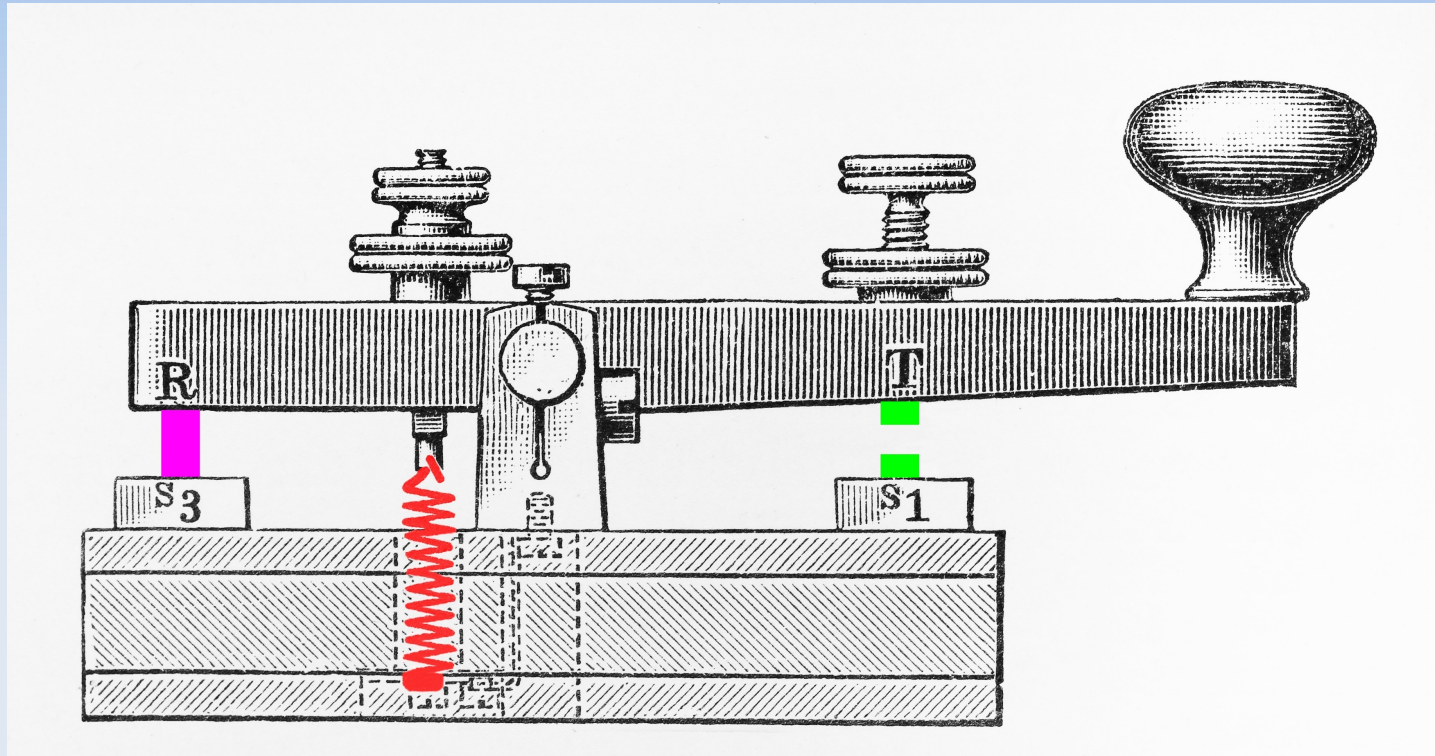
Communication Principles



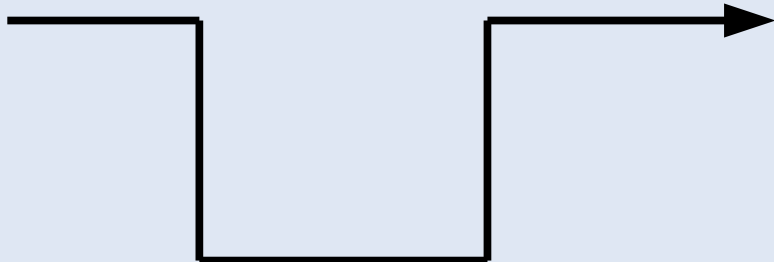
Communication Principles



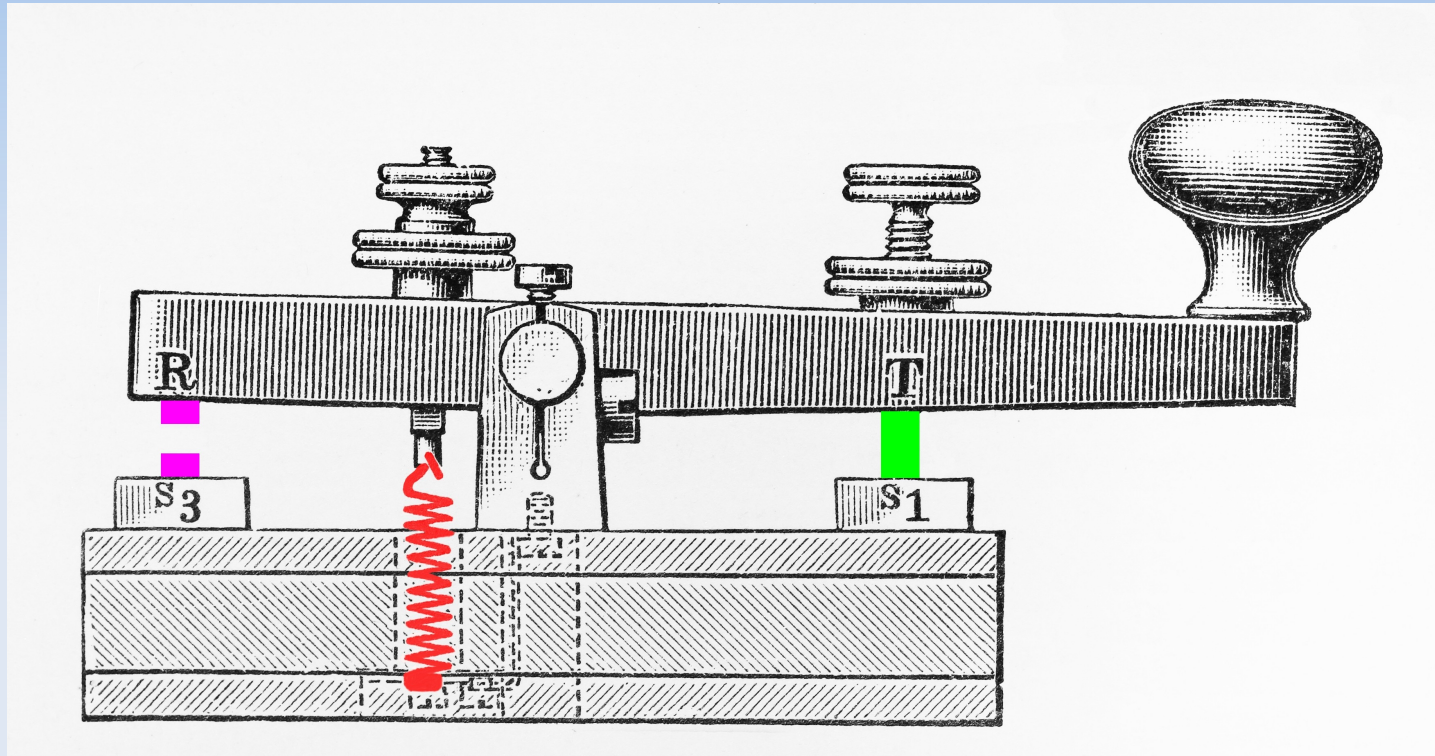
Communication Principles



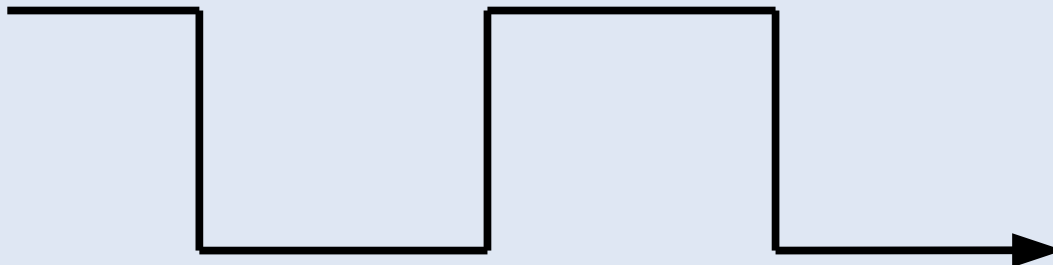
DOT



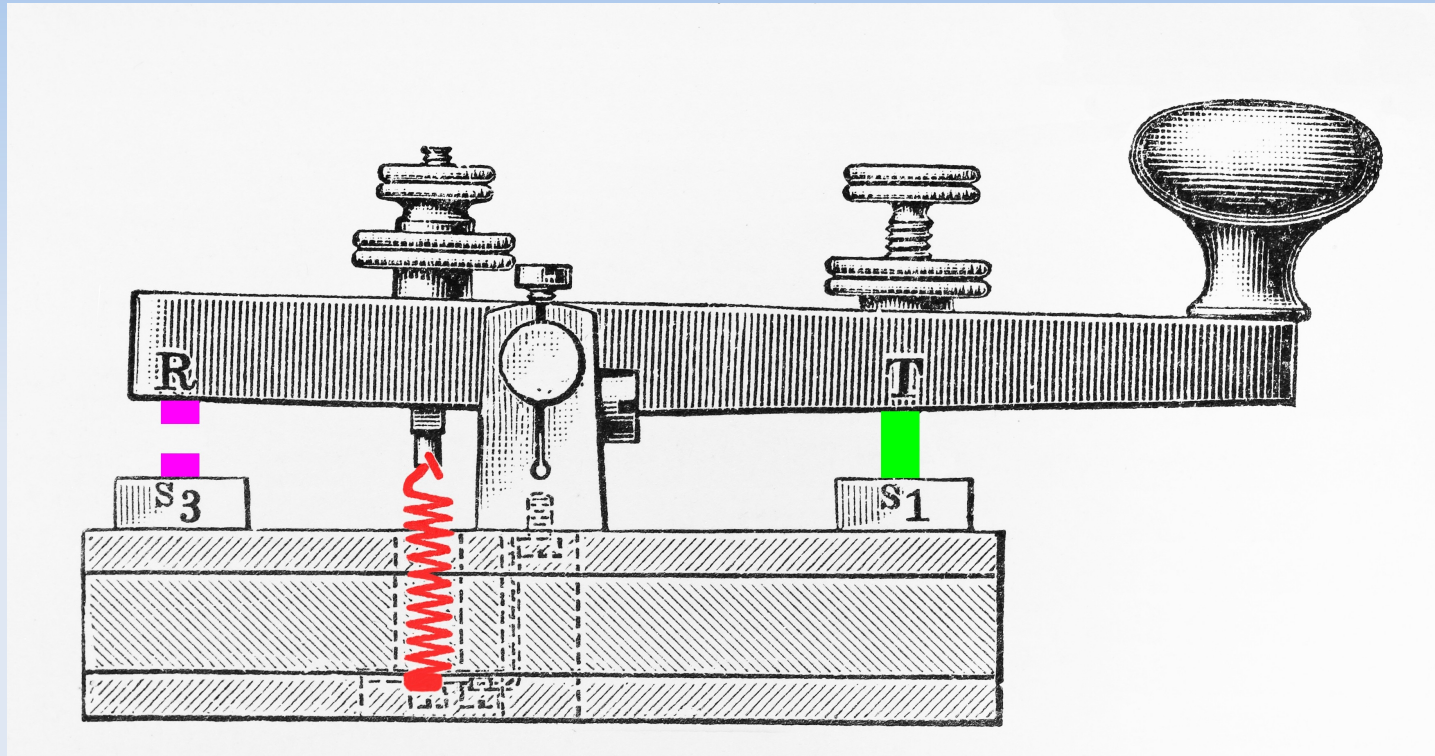
Communication Principles



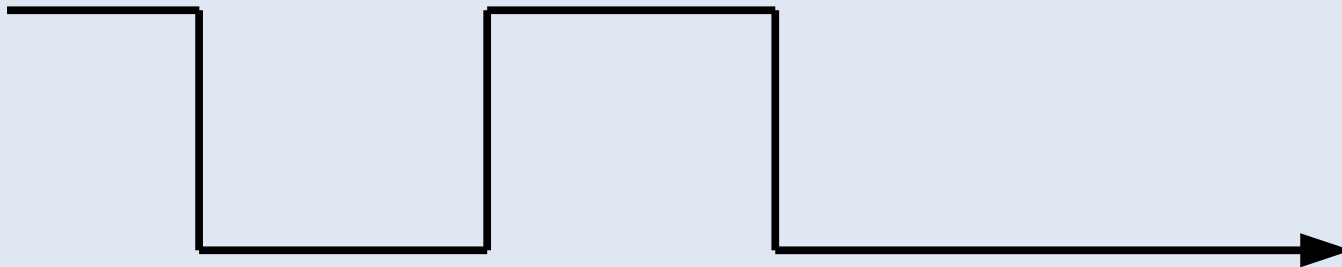
DOT



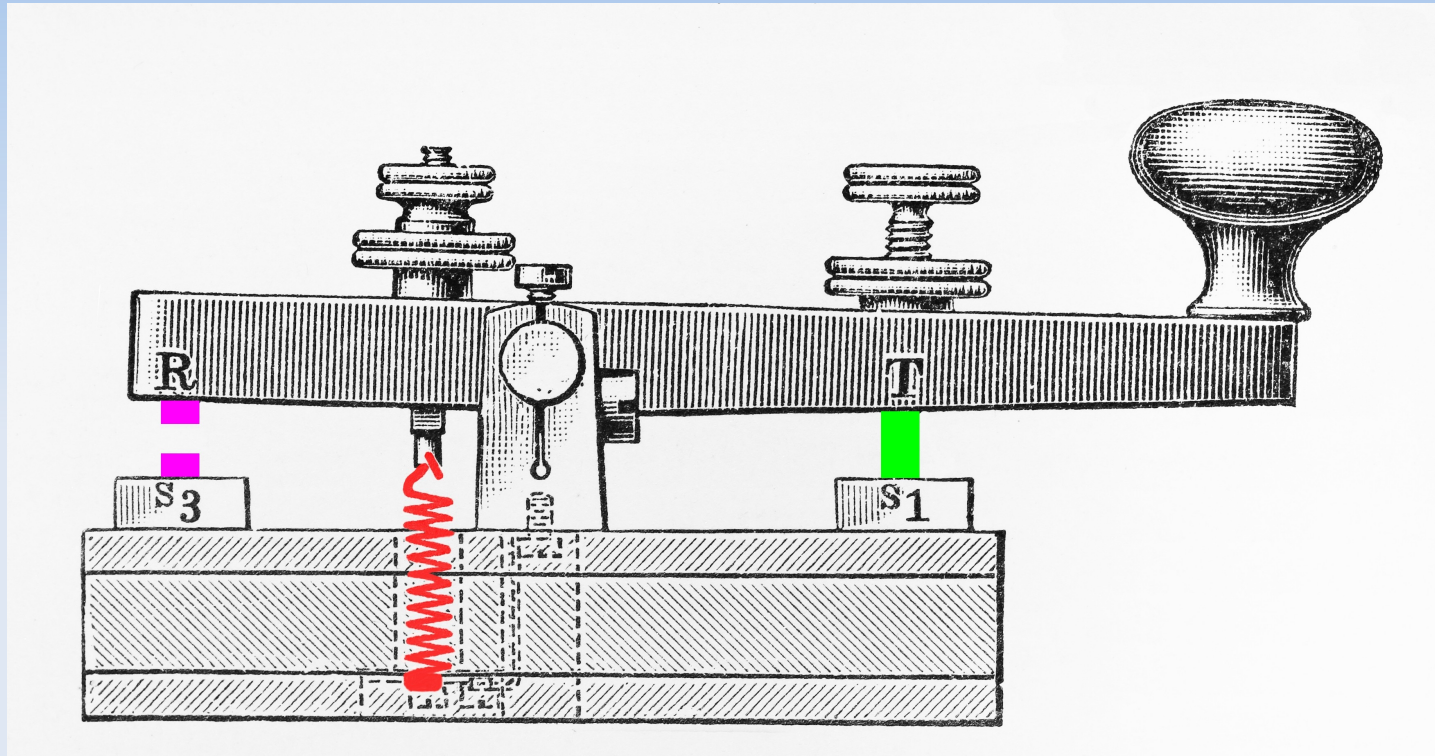
Communication Principles



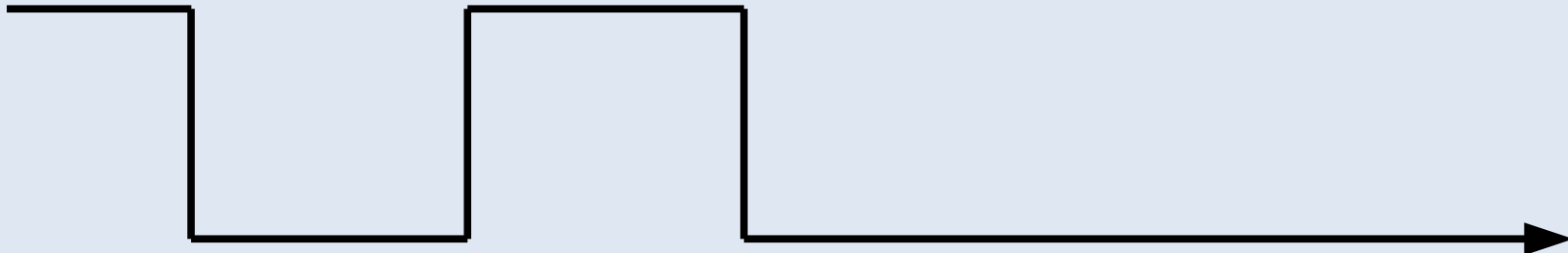
DOT



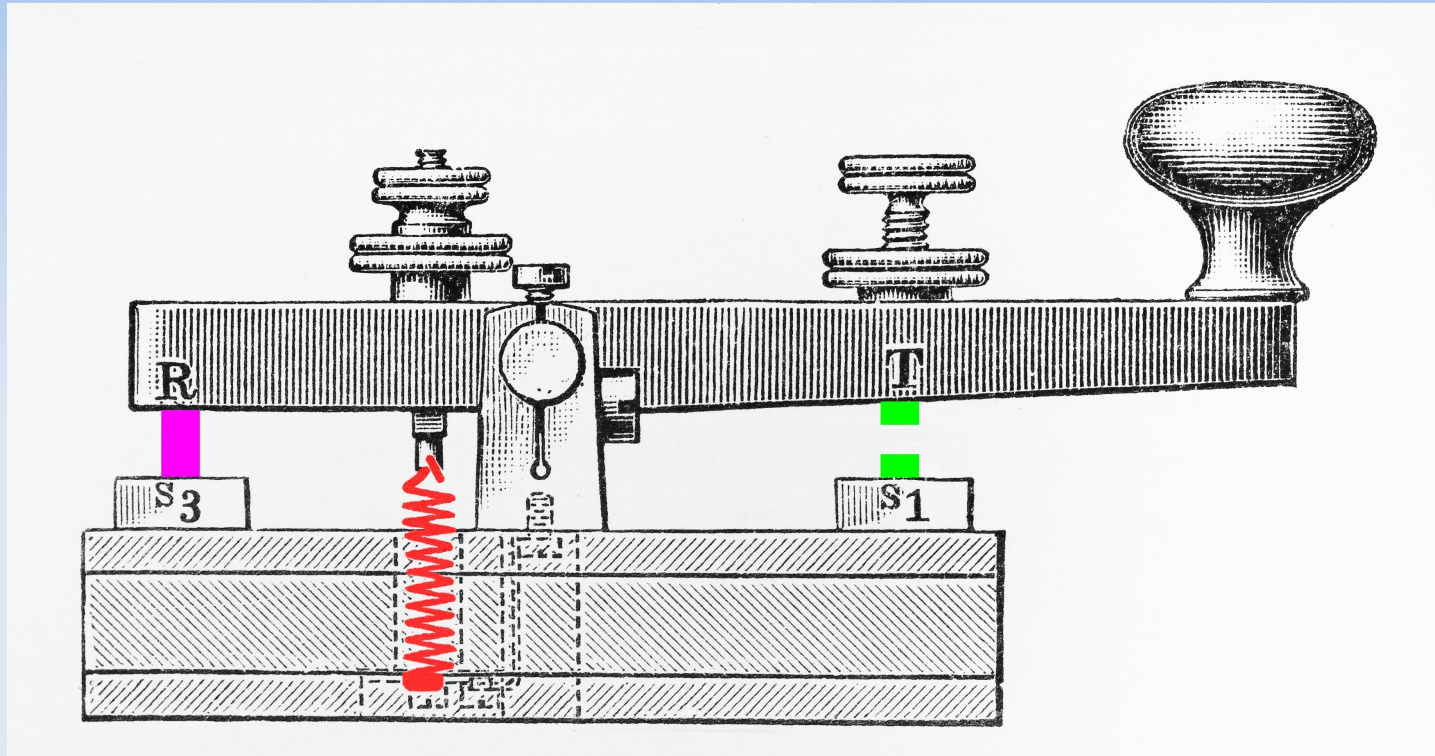
Communication Principles



DOT

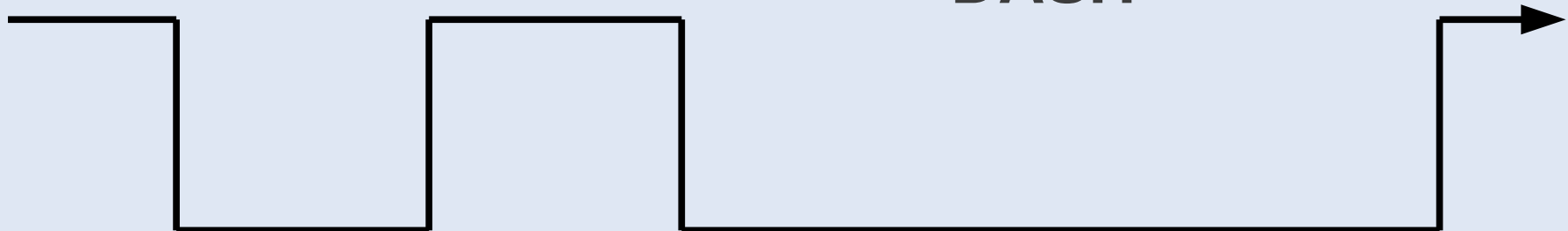


Communication Principles

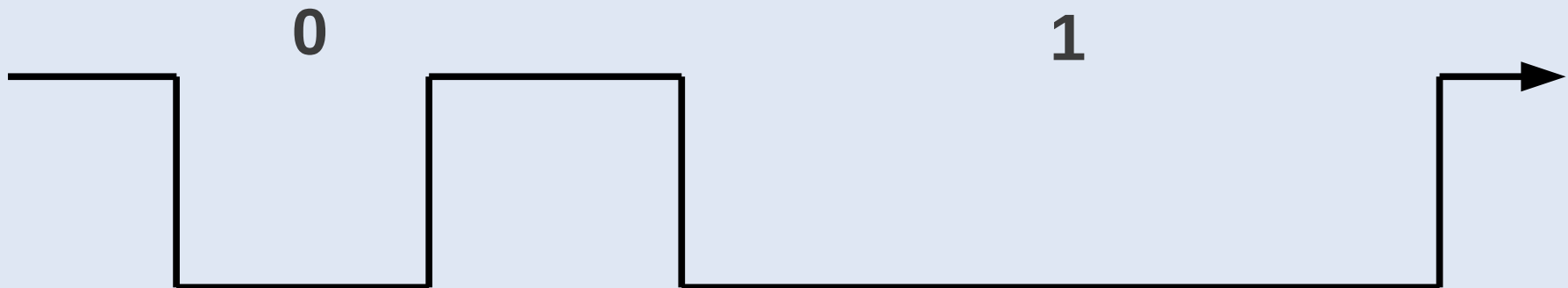
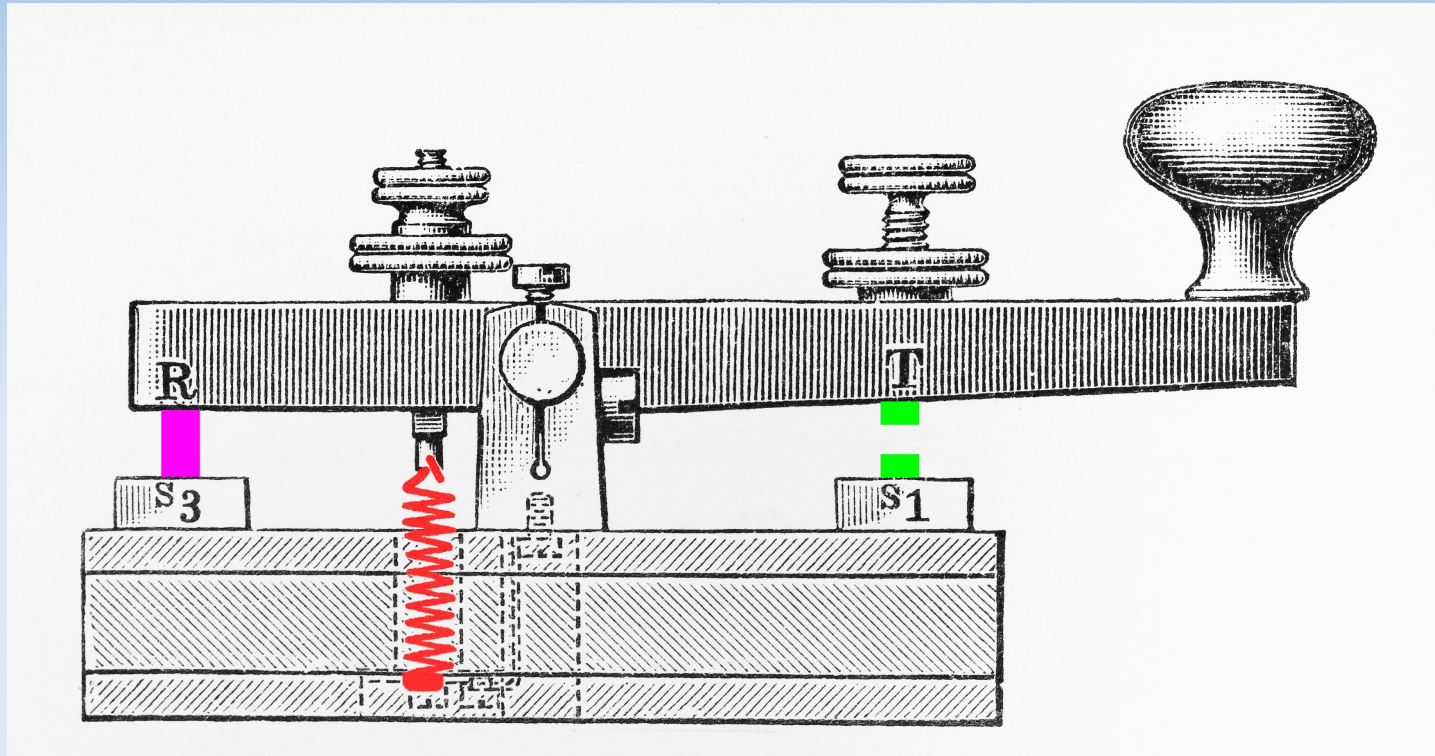


DOT

DASH



Communication Principles



Communication Principles

- Asynchronous Communication
 - No External Clock Signal
 - Morse Code / Telegraph
 - RS-232/UART

Communication Principles

- Asynchronous Communication
 - No External Clock Signal
 - Morse Code / Telegraph
 - RS-232/UART
 - Universal
 - Asynchronous
 - Receiver
 - Transmitter

Communication Principles

- Asynchronous Communication
 - No External Clock Signal
 - Morse Code / Telegraph
 - RS-232/UART
 - Agreed Upon Period Length
 - DOT / DASH
 - Baud Rate

Communication Principles

- Asynchronous Communication
 - No External Clock Signal
 - Morse Code / Telegraph
 - RS-232/UART
 - Agreed Upon Period Length
 - Accurate Timing Device

Communication Principles

- Asynchronous Communication
 - No External Clock Signal
 - Morse Code / Telegraph
 - RS-232/UART
 - Agreed Upon Period Length
 - Accurate Timing Device
 - Crystals
 - Oscillators
 - System Clock Dividers
 - Atmel AVR with 10MHz clock
 - $10\text{MHz} / 20 / 4 = 125000$
 - $115200 \text{ vs. } 125000 = 8.5\% \text{ error}$

Communication Principles

- Asynchronous Communication
- The Problem



Communication Principles

- Asynchronous Communication
- The Problem
 - Simple peripherals
 - Reduced external components
 - No need to set period length
 - Wide range of frequencies



Communication Principles

- Asynchronous Communication
- The Problem
- Synchronous Communication

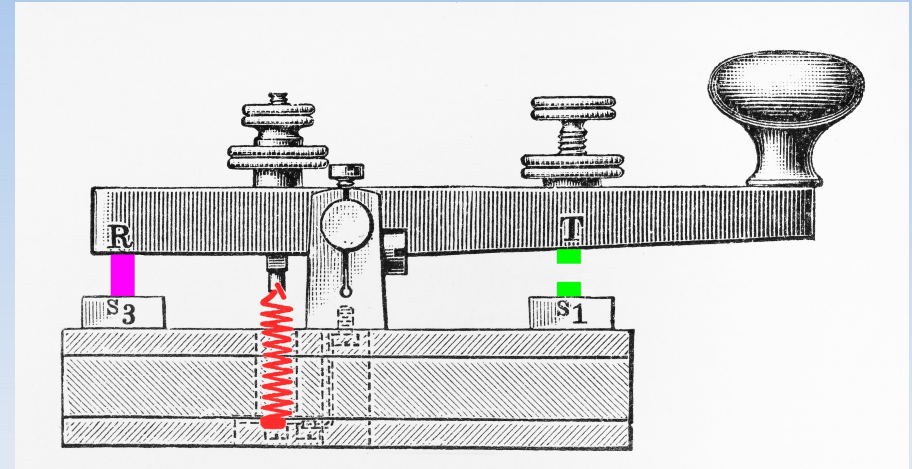
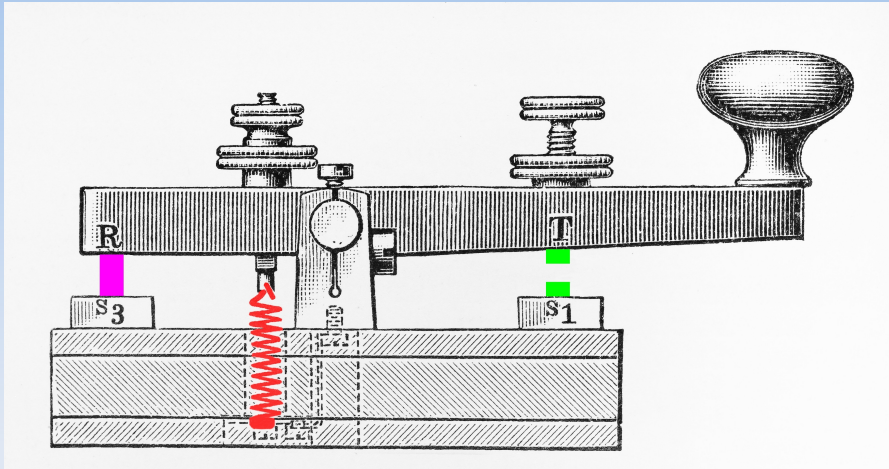
Communication Principles

- Asynchronous Communication
- The Problem
- Synchronous Communication
 - Uses Dedicated Clock Signal

Communication Principles

- Asynchronous Communication
- The Problem
- Synchronous Communication
 - Uses Dedicated Clock Signal
 - Edison Stock Quotes

Communication Principles

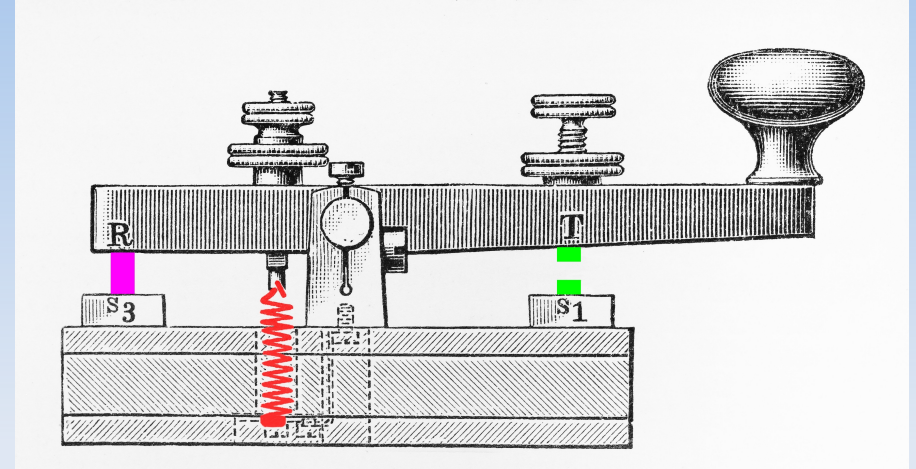
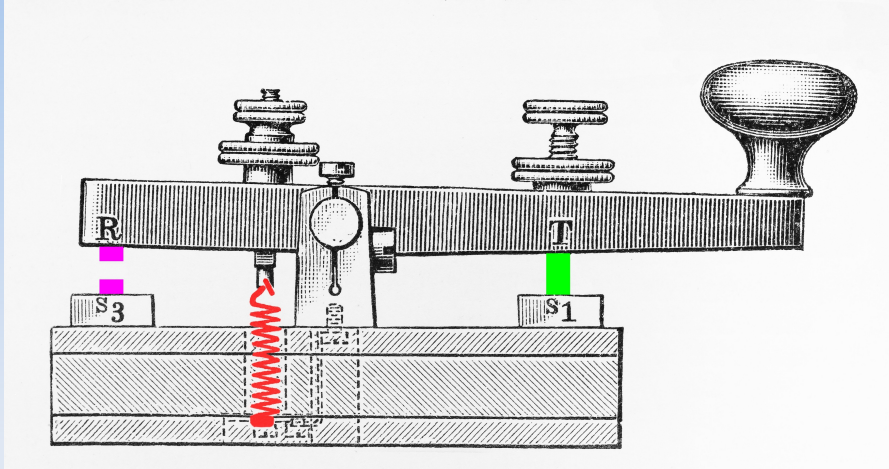


CLOCK



DATA

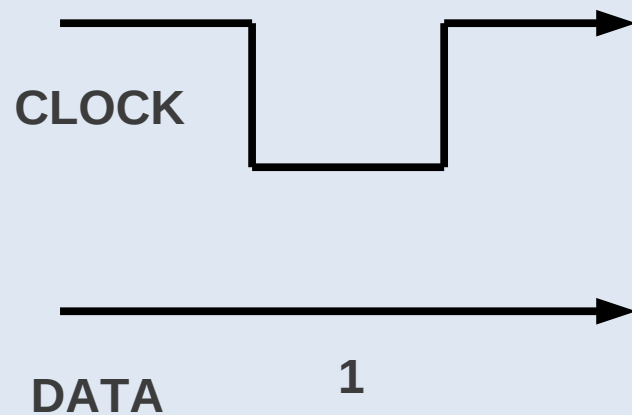
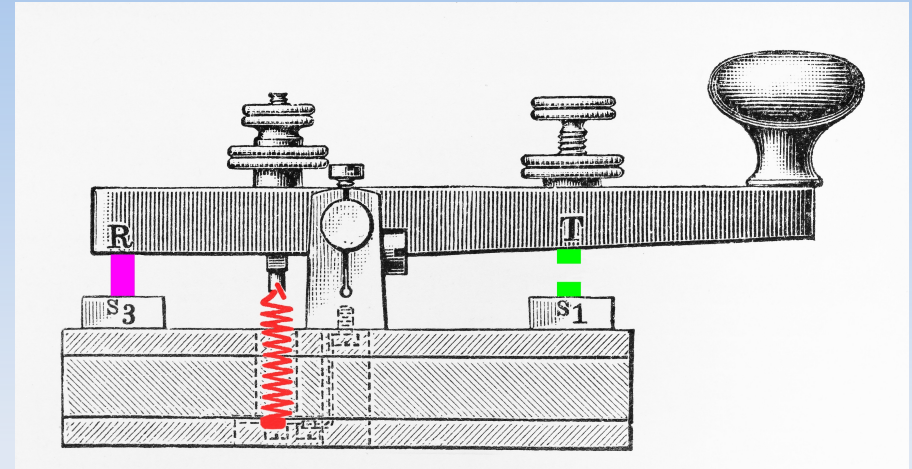
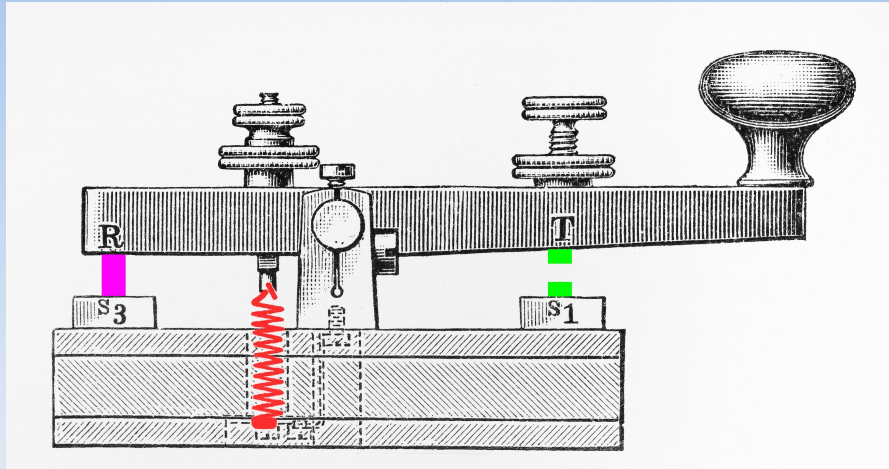
Communication Principles



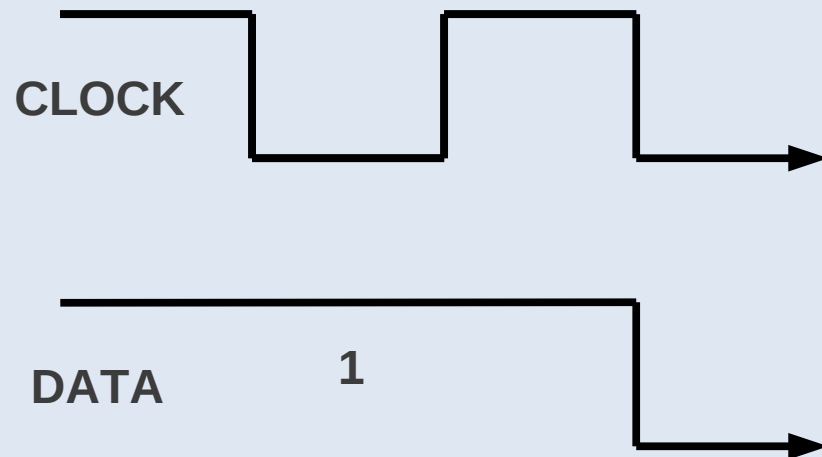
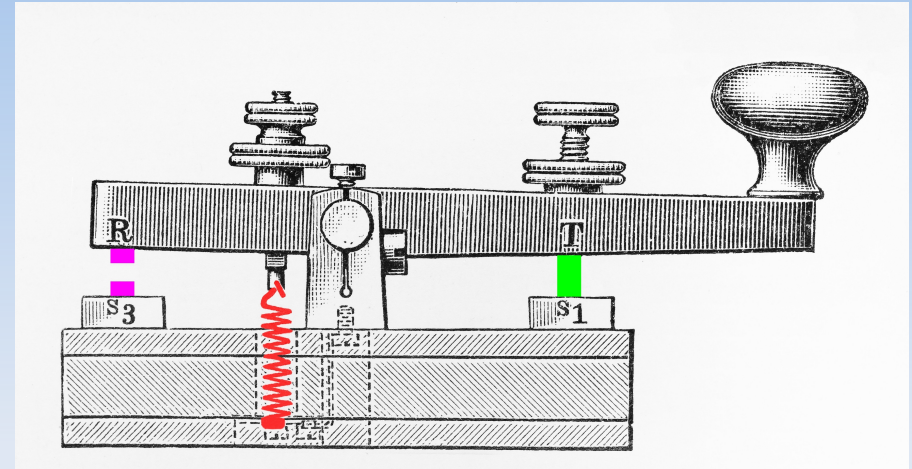
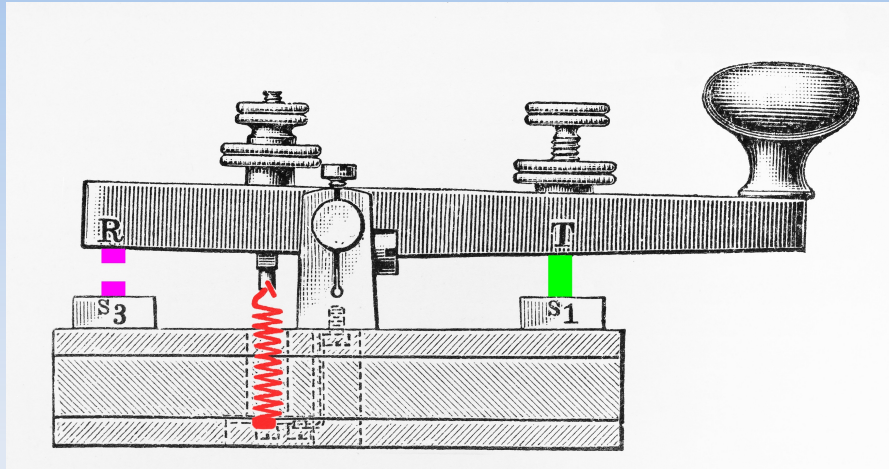
CLOCK

DATA

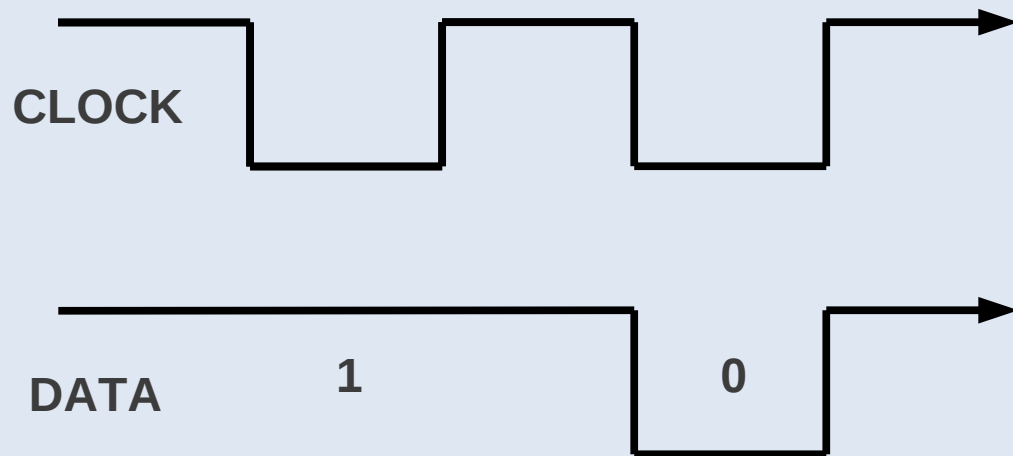
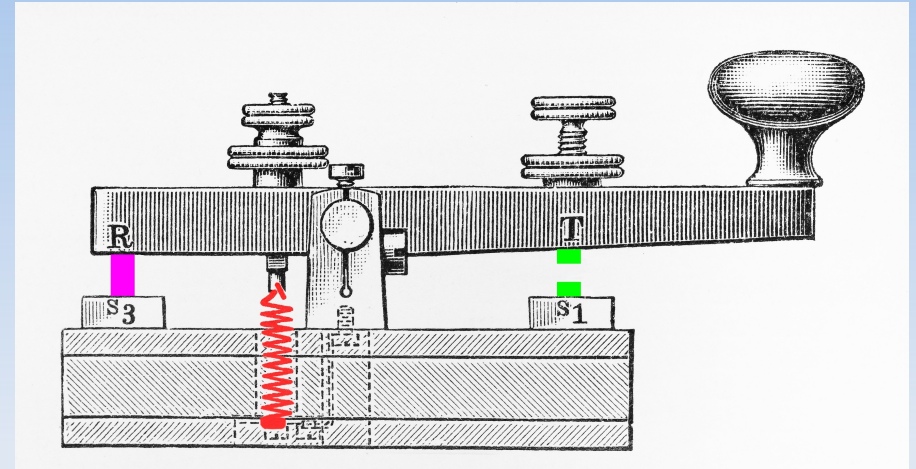
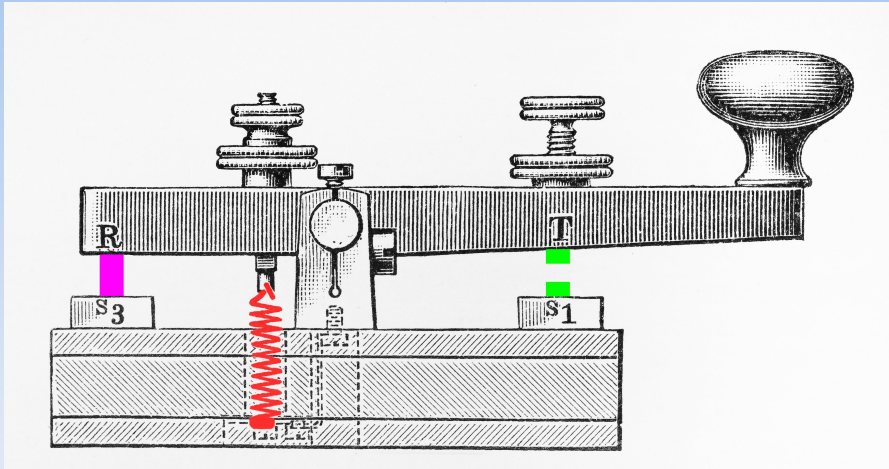
Communication Principles



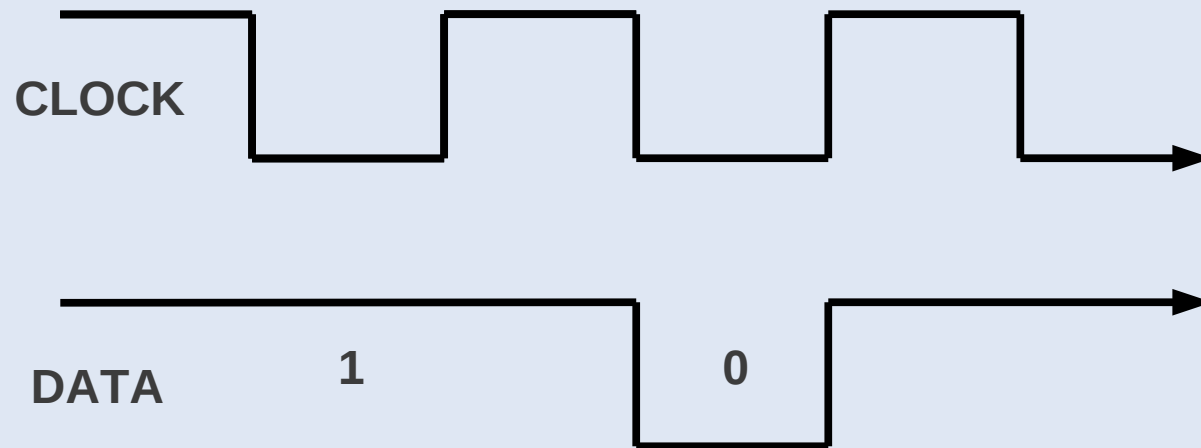
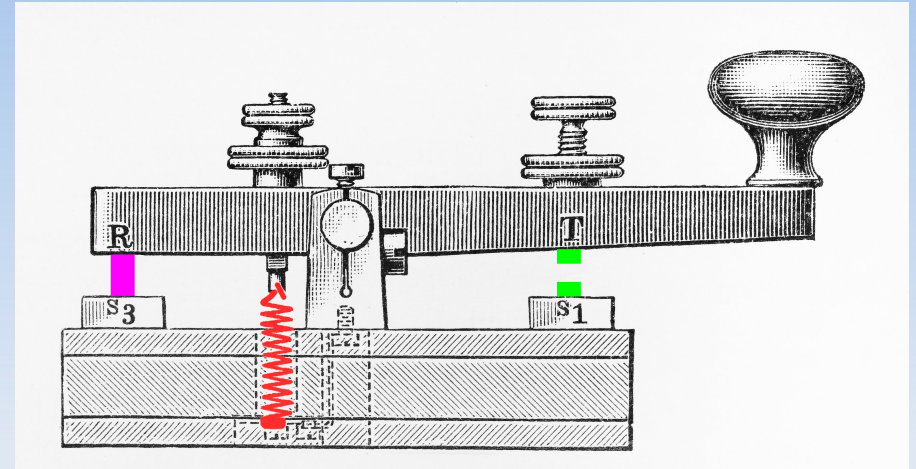
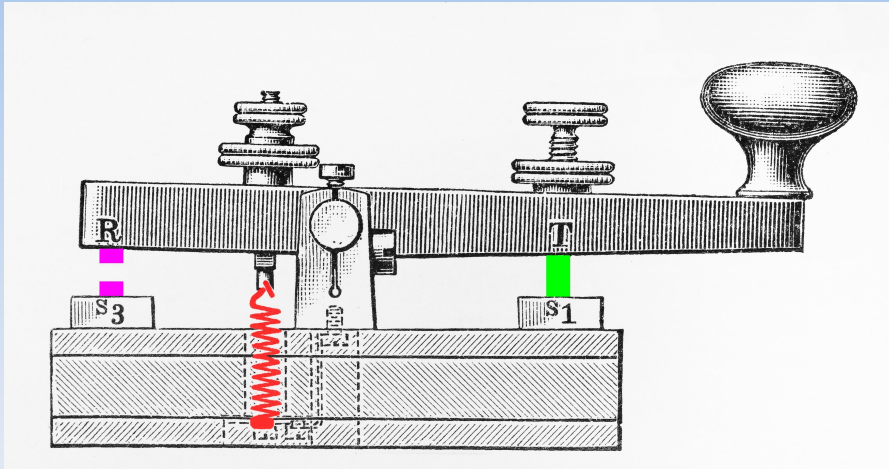
Communication Principles



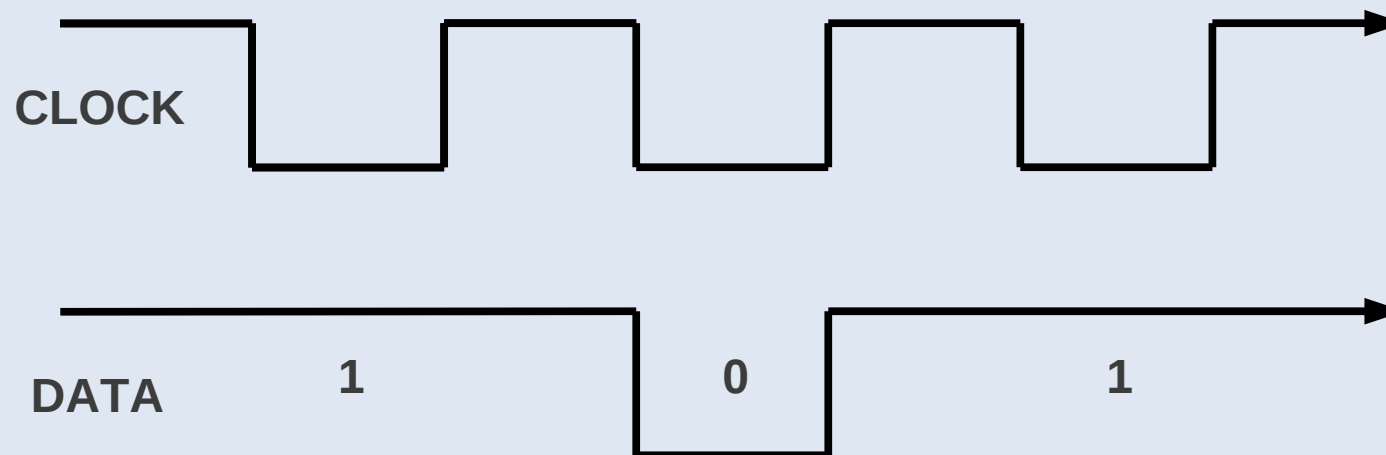
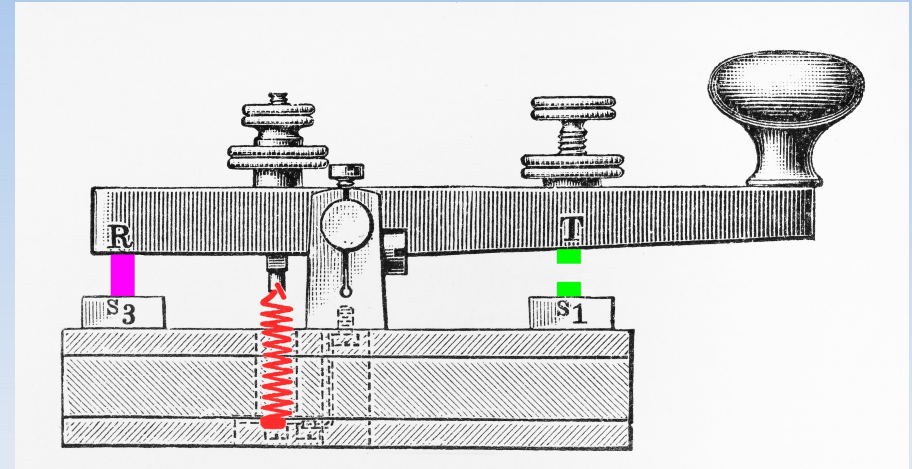
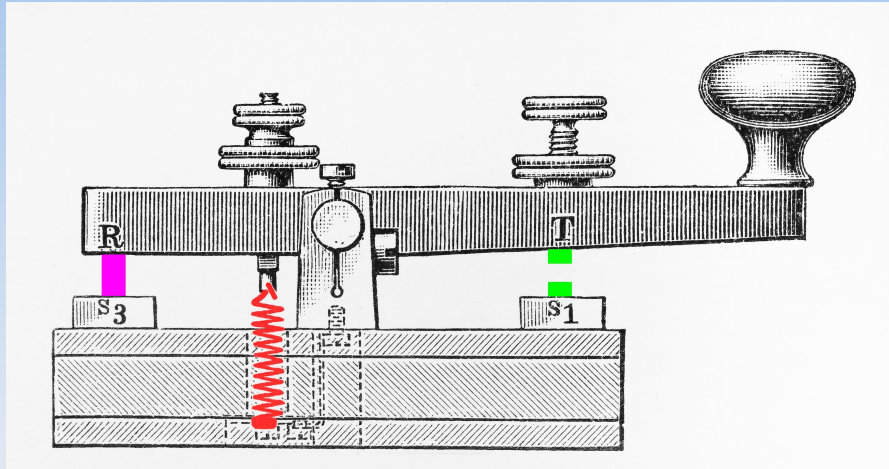
Communication Principles



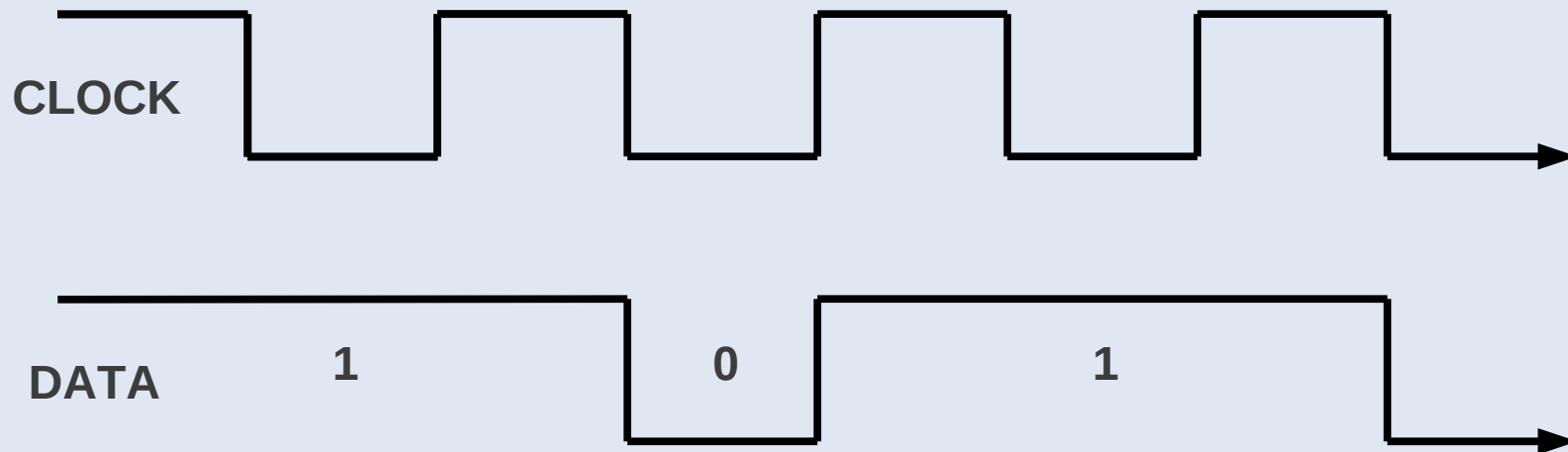
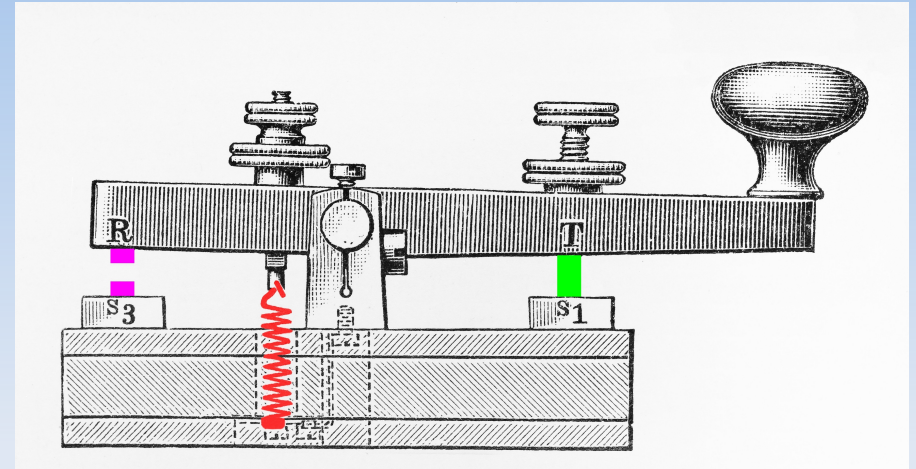
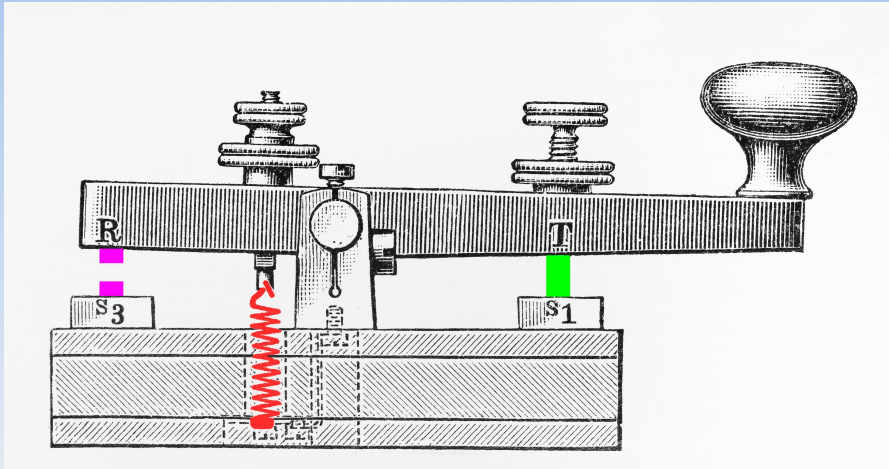
Communication Principles



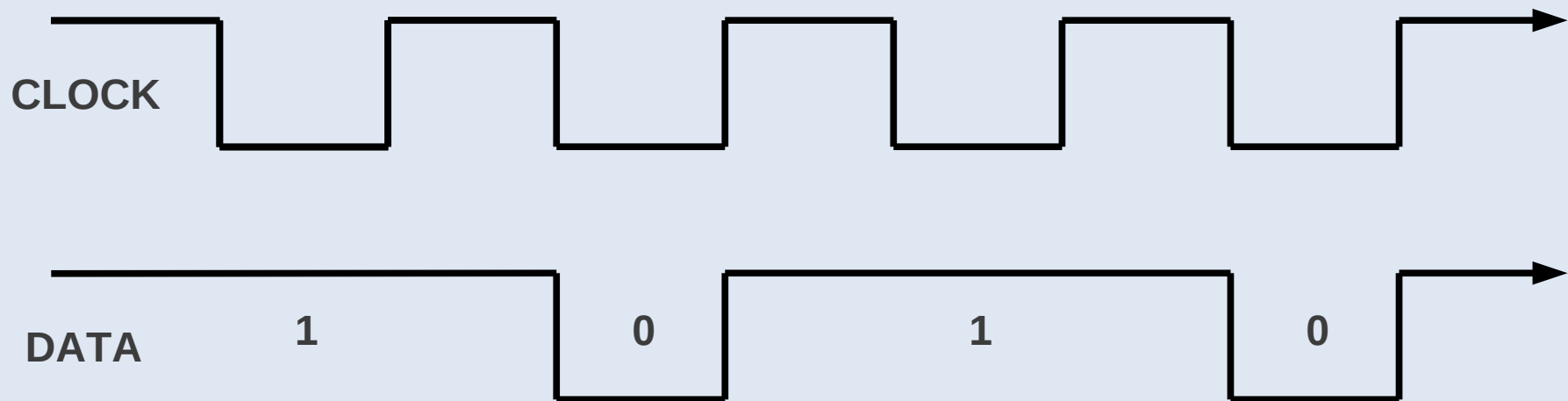
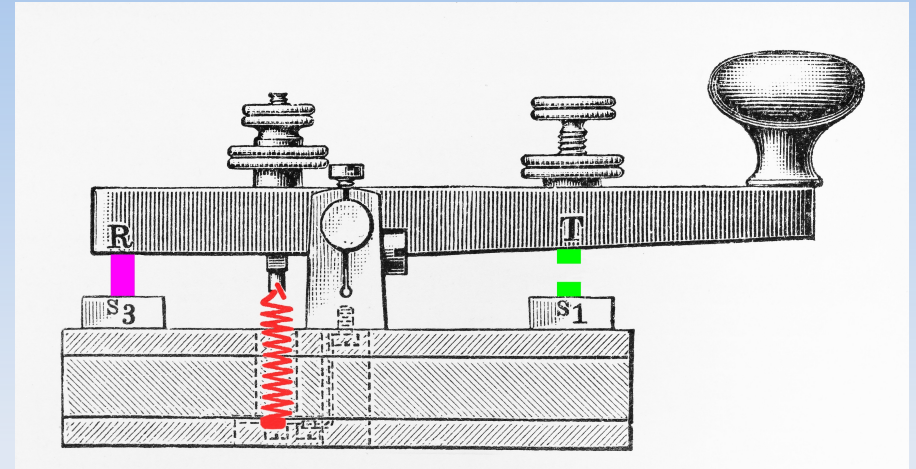
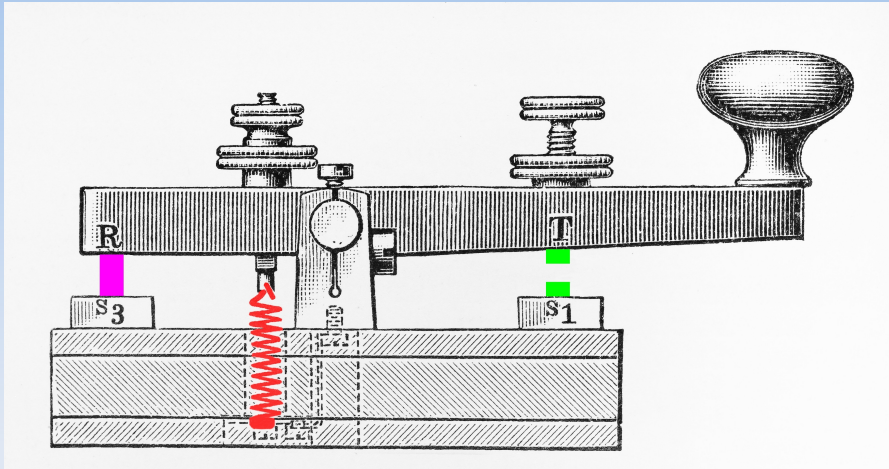
Communication Principles



Communication Principles

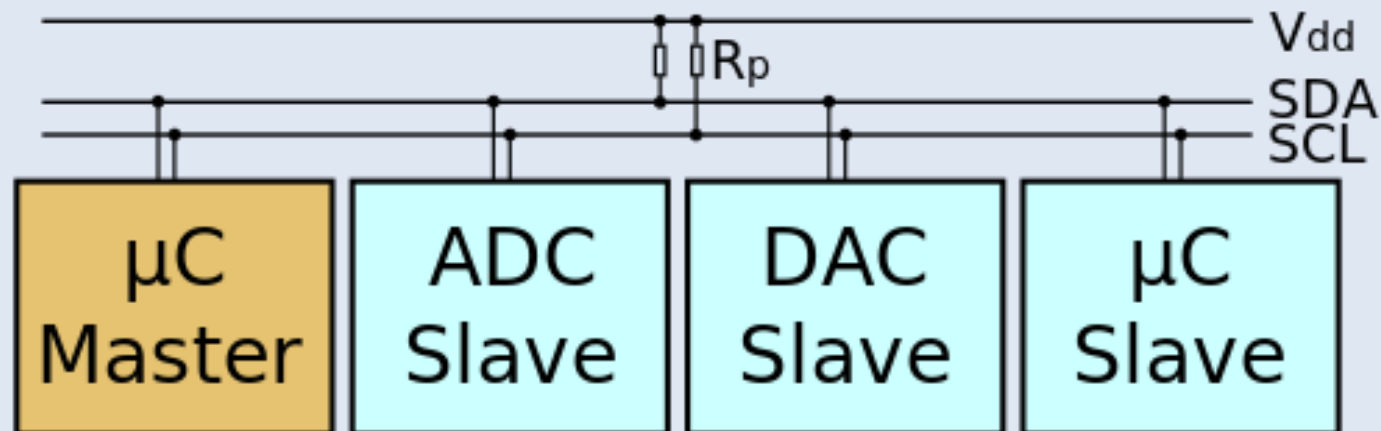


Communication Principles



Communication Principles

- Asynchronous Communication
- The Problem
- Synchronous Communication
 - Uses Dedicated Clock Signal
 - Edison Stock Quotes
 - NXP Developed I2C



Communication Principles

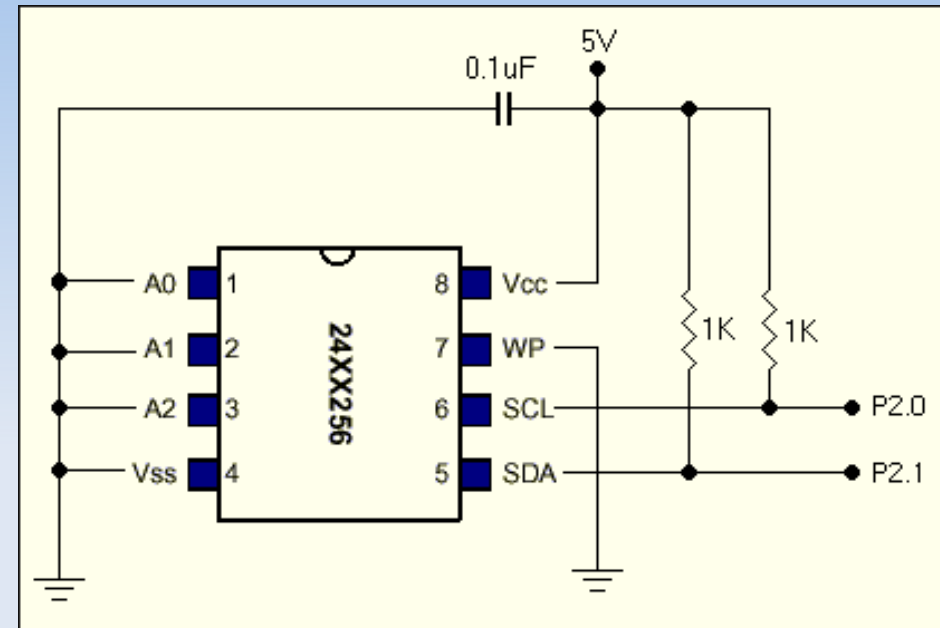
- Asynchronous Communication
- The Problem
- Synchronous Communication
 - Uses Dedicated Clock Signal
 - Edison Stock Quotes
 - NXP Developed I2C
 - Intel Refined with SMBus

Interfacing

- Physical Connections

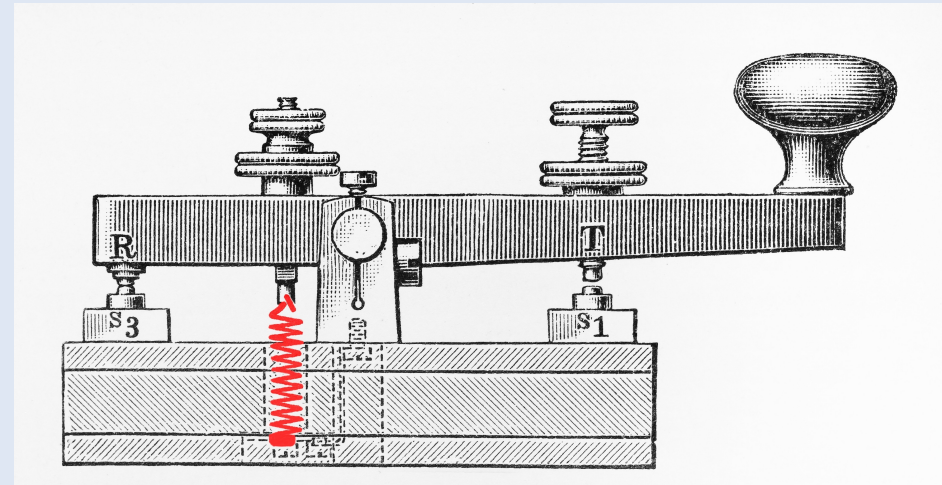
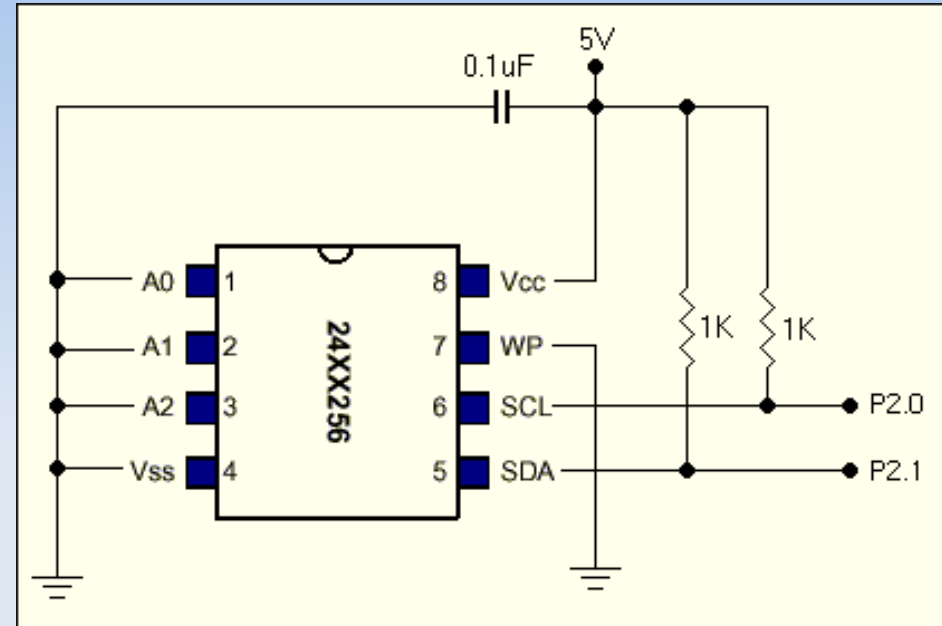
Interfacing

- Physical Connections
 - VCC, SCL, SDA, VSS



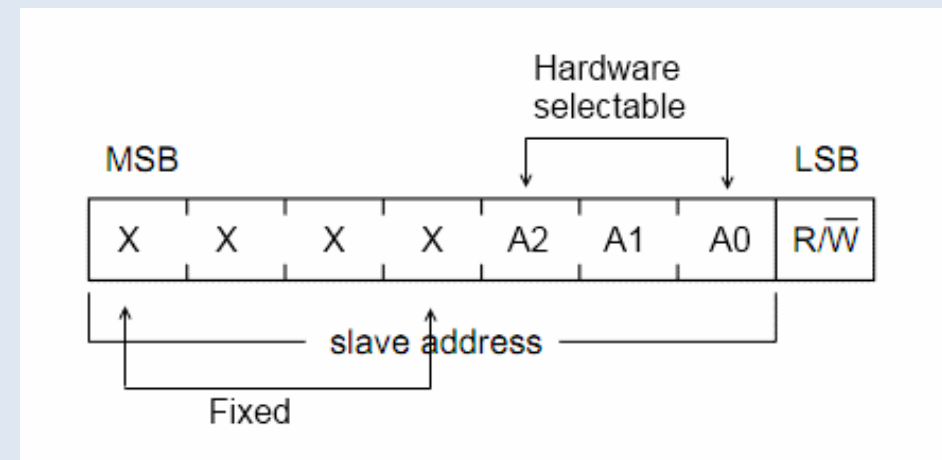
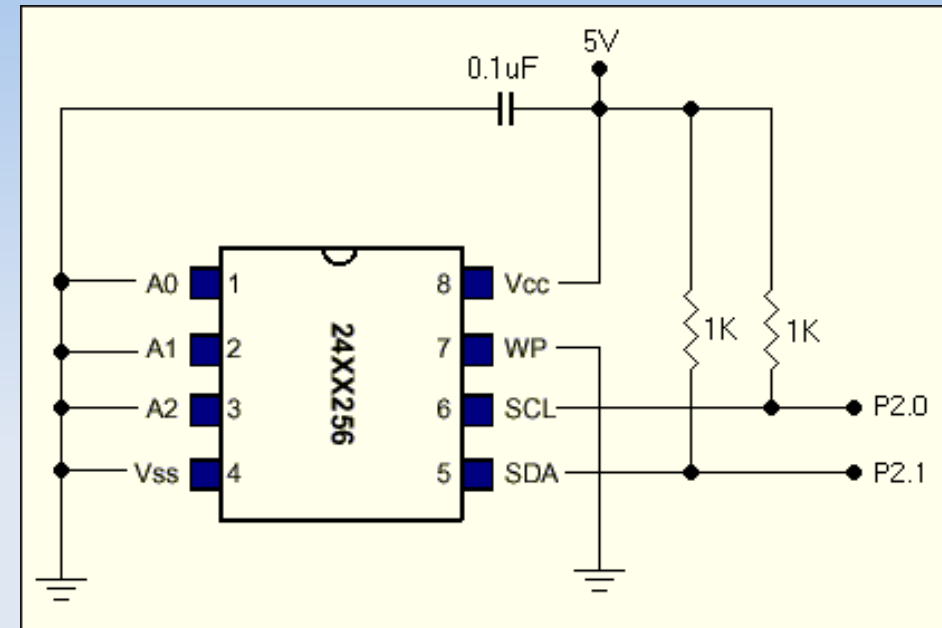
Interfacing

- Physical Connections
 - VCC, SCL, SDA, VSS
 - Pull-Ups



Interfacing

- Physical Connections
 - VCC, SCL, SDA, VSS
 - Pull-Ups
 - Address



Interfacing

- Physical Connections

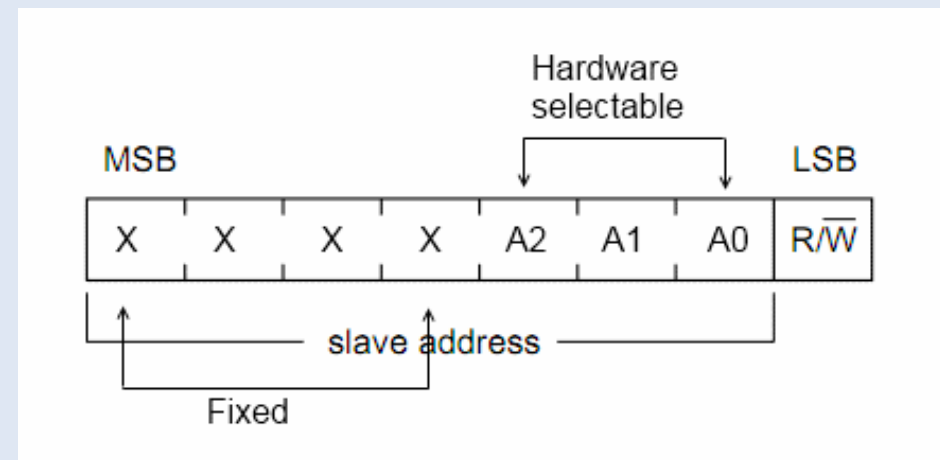
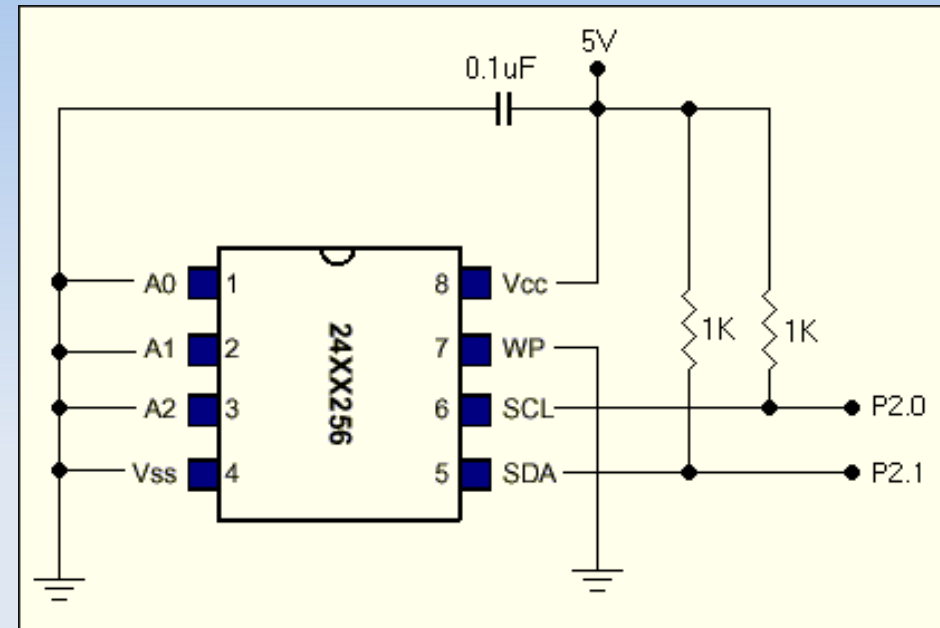
- VCC, SCL, SDA, VSS

- Pull-Ups

- Address

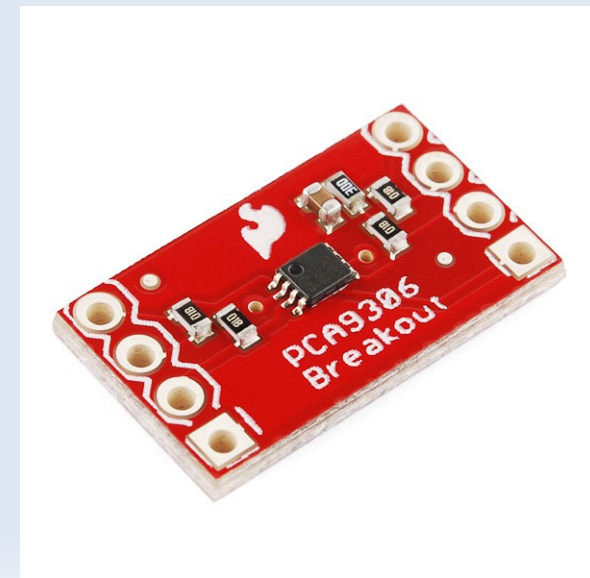
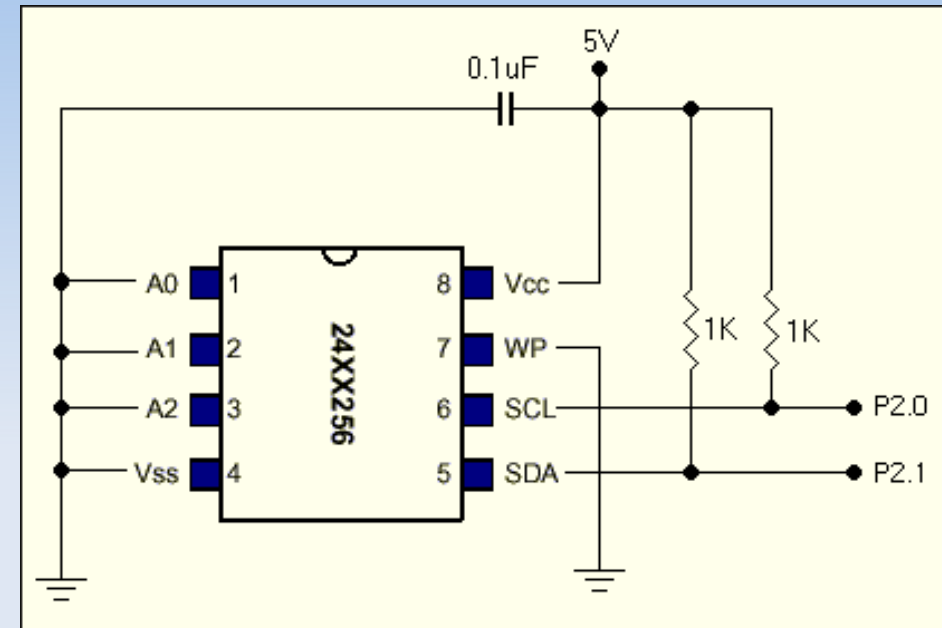
- 7-bits

- LSB Read/Write



Interfacing

- Physical Connections
 - VCC, SCL, SDA, VSS
 - Pull-Ups
 - Address
 - Level Shifters



Interfacing

- Physical Connections
- Drivers

Interfacing

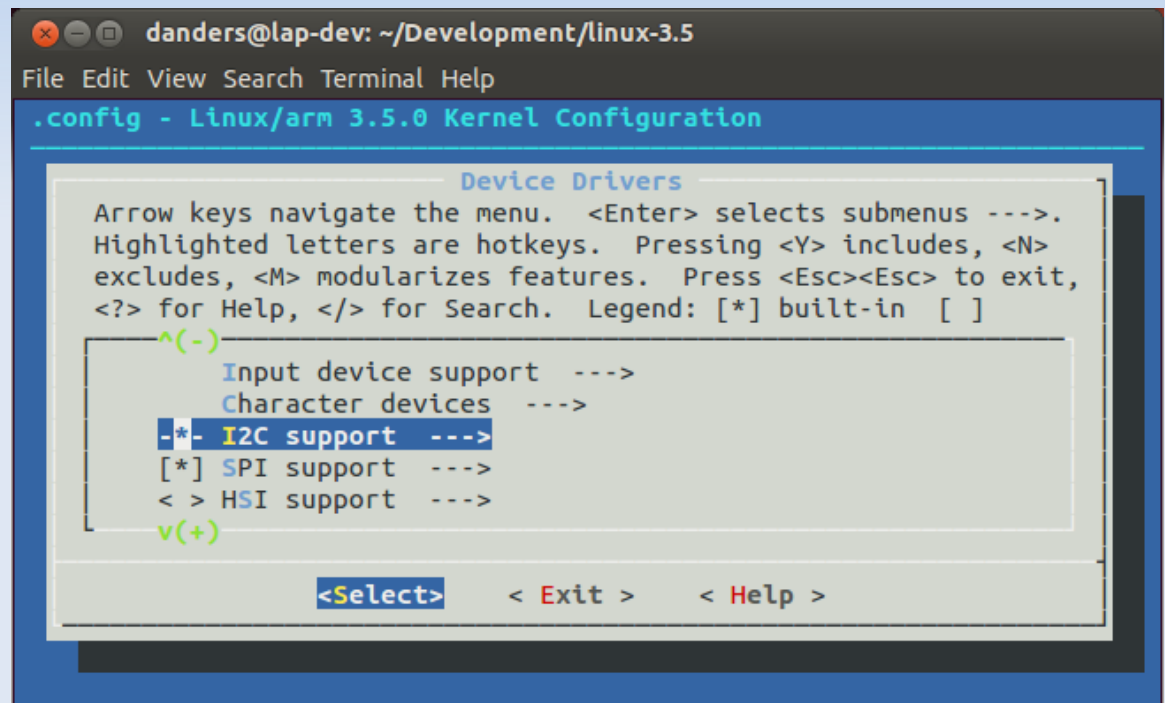
- Physical Connections
- Drivers
 - Bootloaders

Interfacing

- Physical Connections
- Drivers
 - Bootloaders
 - Linux Kernel

Interfacing

- Physical Connections
- Drivers
 - Bootloaders
 - Linux Kernel



```
danders@lap-dev: ~/Development/linux-3.5
File Edit View Search Terminal Help
.config - Linux/arm 3.5.0 Kernel Configuration

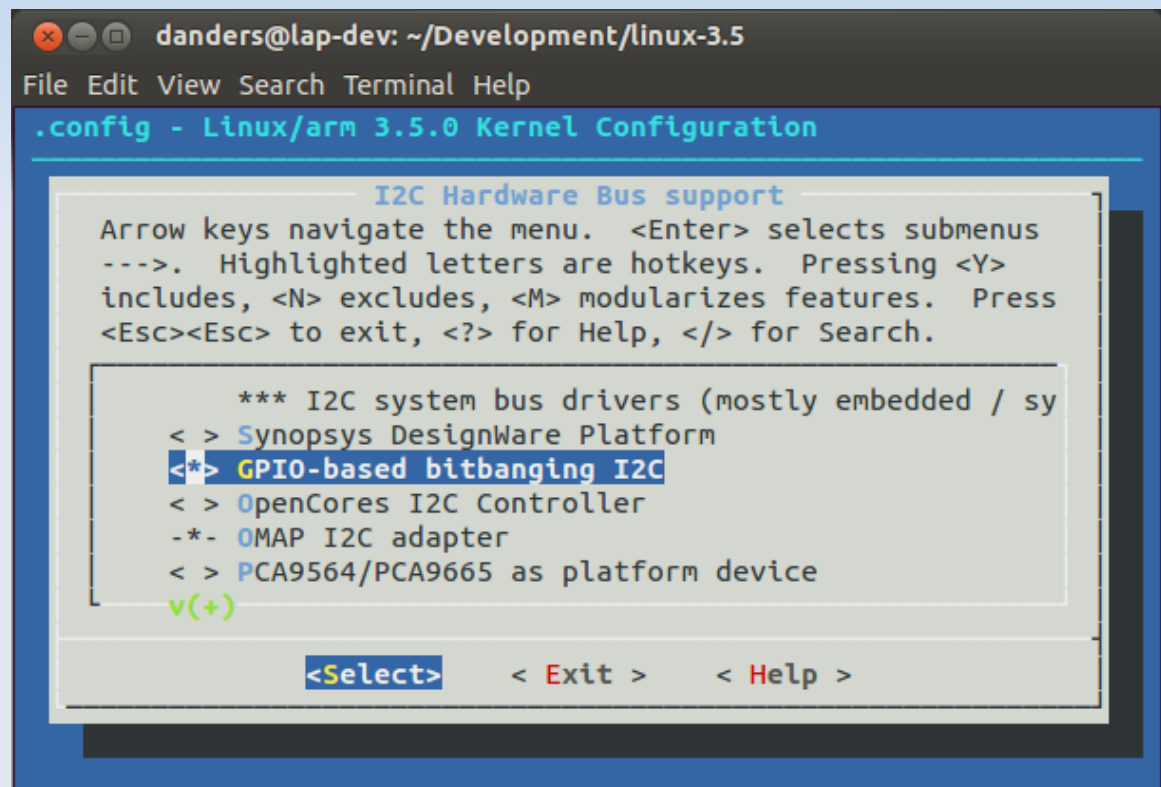
Device Drivers
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N>
excludes, <M> modularizes features. Press <Esc><Esc> to exit,
<?> for Help, </> for Search. Legend: [*] built-in [ ]

^(-)
Input device support --->
Character devices --->
[*] I2C support --->
[*] SPI support --->
< > HSI support --->
v(+)

<Select> < Exit > < Help >
```

Interfacing

- Physical Connections
- Drivers
 - Bootloaders
 - Linux Kernel
 - GPIO Bit-Bang



The screenshot shows a terminal window with the title bar "danders@lap-dev: ~/Development/linux-3.5". The terminal displays the "Linux/arm 3.5.0 Kernel Configuration" menu. The current selection is "I2C Hardware Bus support". The menu text explains navigation: "Arrow keys navigate the menu. <Enter> selects submenus --->. Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search." The list of options under "I2C Hardware Bus support" is: "*** I2C system bus drivers (mostly embedded / sy", "< > Synopsys DesignWare Platform", "<*> GPIO-based bitbanging I2C" (highlighted), "< > OpenCores I2C Controller", "-*- OMAP I2C adapter", and "< > PCA9564/PCA9665 as platform device". At the bottom, there are navigation options: "<Select>", "< Exit >", and "< Help >".

```
danders@lap-dev: ~/Development/linux-3.5
File Edit View Search Terminal Help
.config - Linux/arm 3.5.0 Kernel Configuration

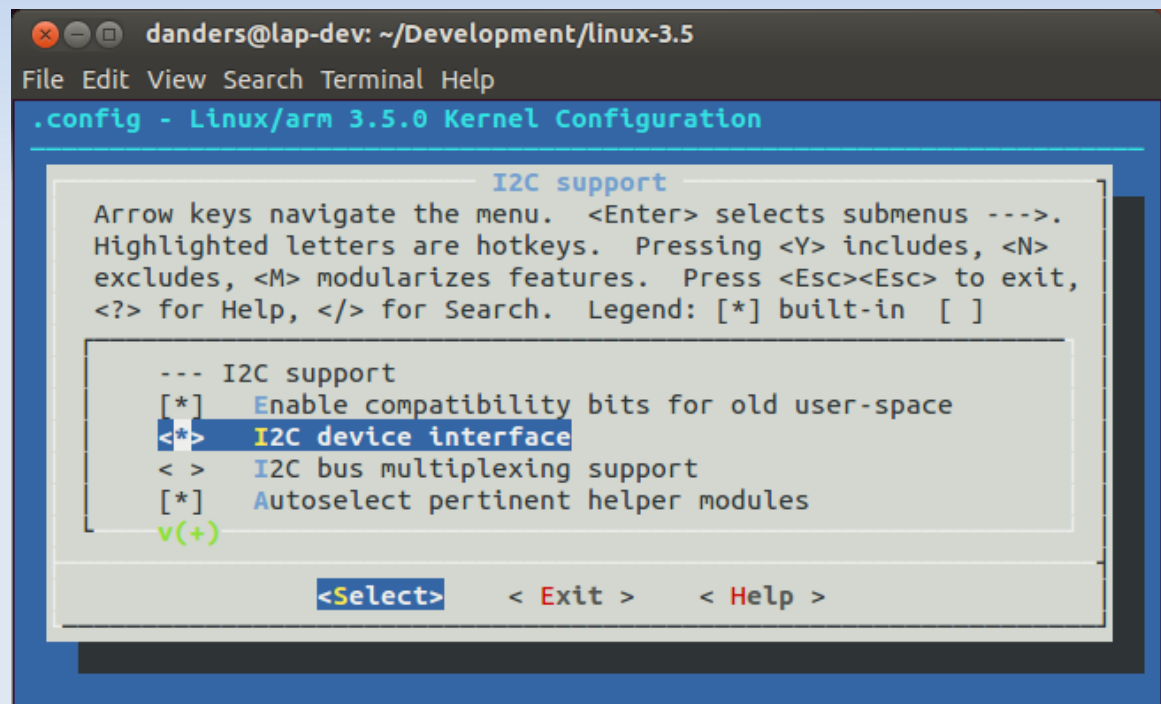
I2C Hardware Bus support
Arrow keys navigate the menu. <Enter> selects submenus
--->. Highlighted letters are hotkeys. Pressing <Y>
includes, <N> excludes, <M> modularizes features. Press
<Esc><Esc> to exit, <?> for Help, </> for Search.

*** I2C system bus drivers (mostly embedded / sy
< > Synopsys DesignWare Platform
<*> GPIO-based bitbanging I2C
< > OpenCores I2C Controller
-*- OMAP I2C adapter
< > PCA9564/PCA9665 as platform device
v(+)

<Select> < Exit > < Help >
```

Interfacing

- Physical Connections
- Drivers
 - Bootloaders
 - Linux Kernel
 - GPIO Bit-Bang
 - I2C CharDev



The screenshot shows a terminal window titled "danders@lap-dev: ~/Development/linux-3.5". The terminal displays the ".config - Linux/arm 3.5.0 Kernel Configuration" menu. The "I2C support" submenu is open, showing the following options:

- I2C support
- [*] Enable compatibility bits for old user-space
- <*> I2C device interface (highlighted)
- < > I2C bus multiplexing support
- [*] Autoselect pertinent helper modules

At the bottom of the menu, there are three buttons: "<Select>", "< Exit >", and "< Help >".

Interfacing

- Physical Connections
- Drivers
- I2C Tools
 - i2cdetect
 - i2cdump

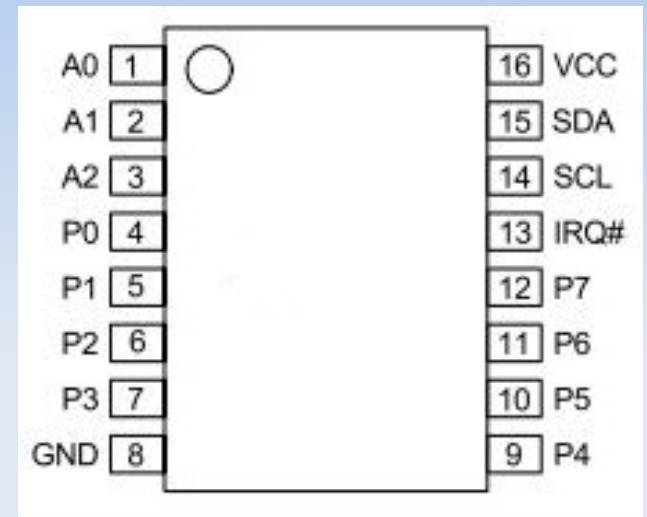
[illegible]

Board Bringup

- I2C GPIO Expanders

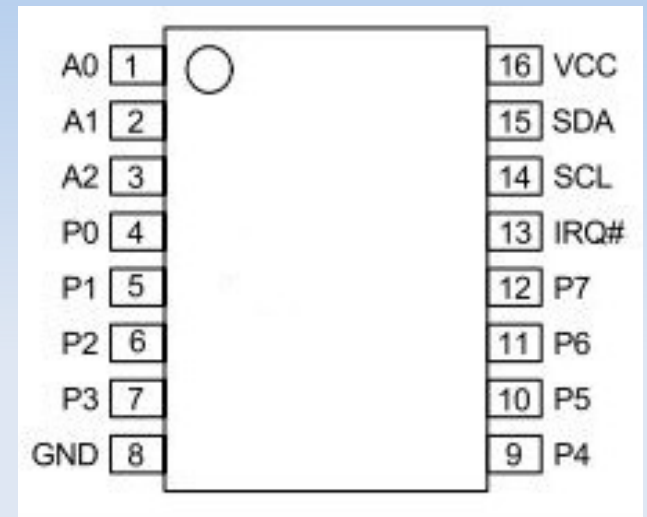
Board Bringup

- I2C GPIO Expanders
 - Devices



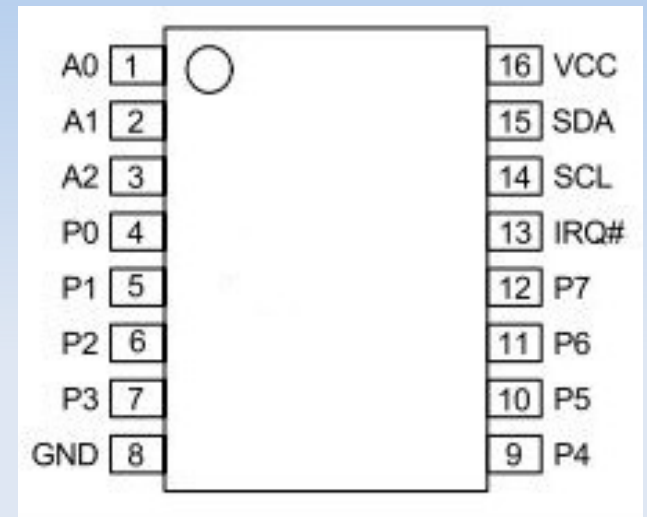
Board Bringup

- I2C GPIO Expanders
 - Devices
 - 4 to 24 Inputs or Output



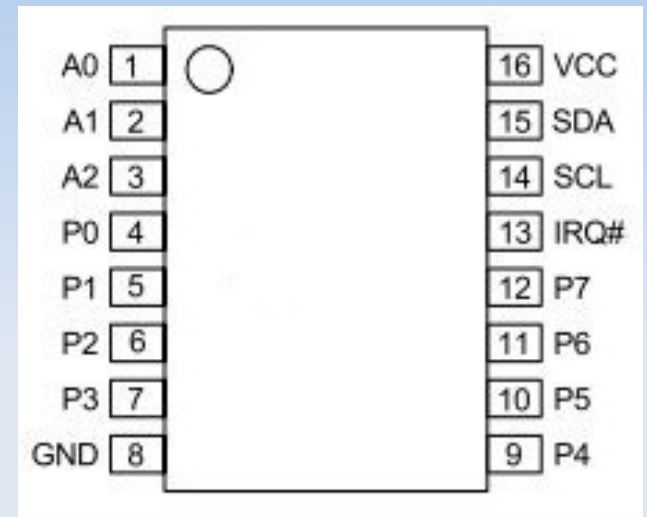
Board Bringup

- I2C GPIO Expanders
 - Devices
 - 4 to 24 Inputs or Output
 - IRQ for input events



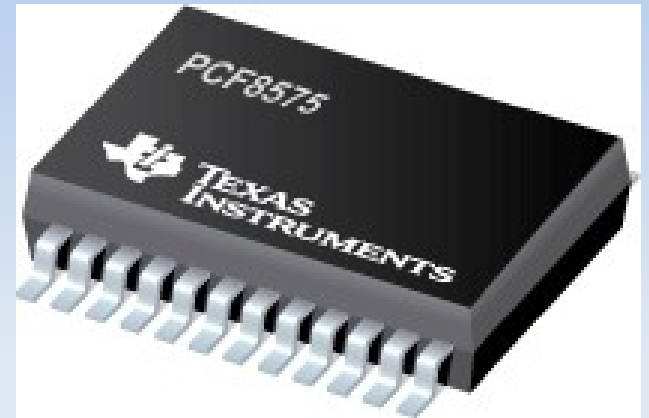
Board Bringup

- I2C GPIO Expanders
 - Devices
 - 4 to 24 Inputs or Output
 - IRQ for input events
 - Voltage range support



Board Bringup

- I2C GPIO Expanders
 - Devices
 - 4 to 24 Inputs or Output
 - IRQ for input events
 - Voltage range support
 - Generic PCF857X



Board Bringup

- I2C GPIO Expanders
 - Devices
 - Retro-fit

Board Bringup

- I2C GPIO Expanders
 - Devices
 - Retro-fit
 - Only needs 2 GPIOS from Host

Board Bringup

- I2C GPIO Expanders
 - Devices
 - Retro-fit
 - Only needs 2 GPIOs from Host



Board Bringup

- I2C GPIO Expanders
 - Devices
 - Retro-fit
 - Only needs 2 GPIOS from Host
 - Different Voltage Levels

Board Bringup

- I2C GPIO Expanders
 - Devices
 - Retro-fit
 - Only needs 2 GPIOS from Host
 - Different Voltage Levels
 - New GPIOs are Transparent

Board Bringup

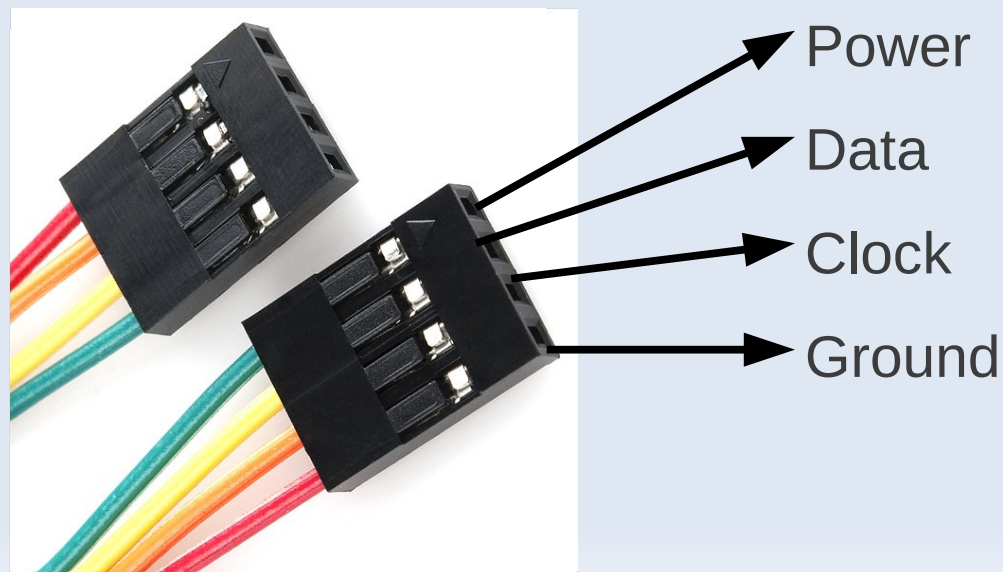
- I2C GPIO Expanders
 - Devices
 - Retro-fit
 - Only needs 2 GPIOS from Host
 - Different Voltage Levels
 - New GPIOs are Transparent
 - Inputs used for versioning

Board Bringup

- I2C GPIO Expanders
 - Devices
 - Retro-fit
 - Debugging

Board Bringup

- I2C GPIO Expanders
 - Devices
 - Retro-fit
 - Debugging
 - Four Wire Connection



Board Bringup

- I2C GPIO Expanders
 - Devices
 - Retro-fit
 - Debugging
 - Four Wire Connection
 - Provide Buttons for Test Modes



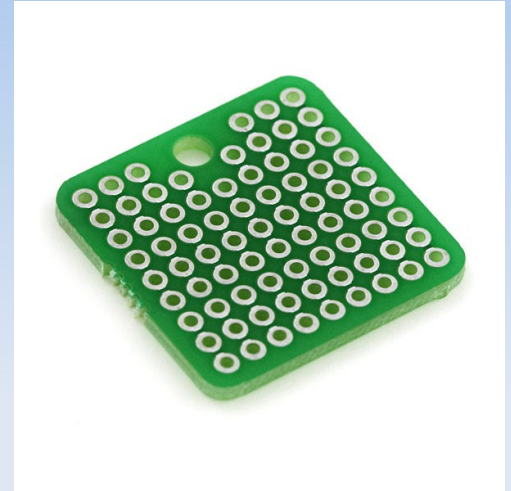
Board Bringup

- I2C GPIO Expanders
 - Devices
 - Retro-fit
 - Debugging
 - Four Wire Connection
 - Provide Buttons for Test Modes
 - Provide LEDS for Low Level Feedback



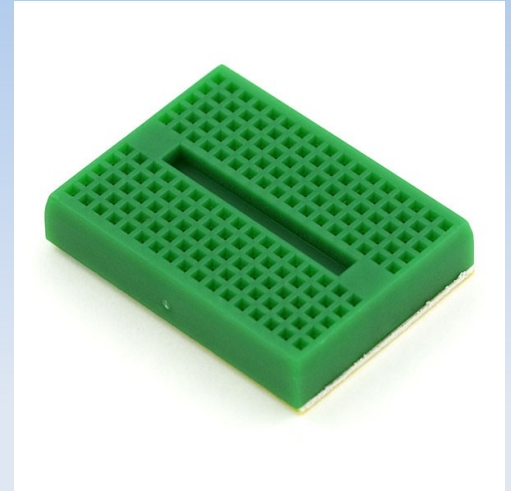
Board Bringup

- I2C GPIO Expanders
 - Devices
 - Retro-fit
 - Debugging
 - Four Wire Connection
 - Provide Buttons for Test Modes
 - Provide LEDS for Low Level Feedback
 - Easily Prototyped



Board Bringup

- I2C GPIO Expanders
 - Devices
 - Retro-fit
 - Debugging
 - Four Wire Connection
 - Provide Buttons for Test Modes
 - Provide LEDS for Low Level Feedback
 - Easily Prototyped

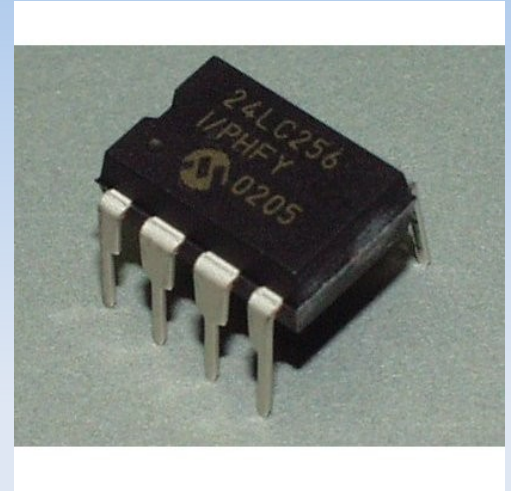


Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS

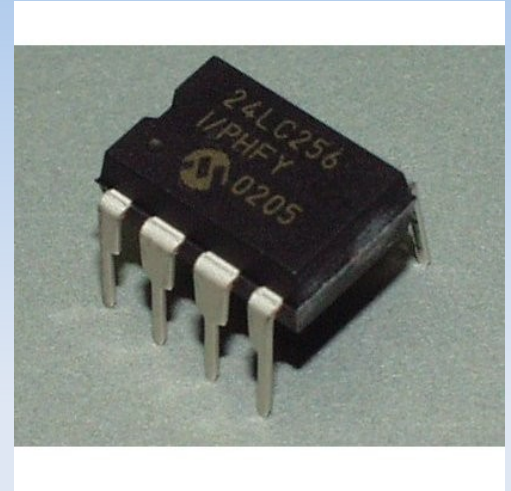
Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices



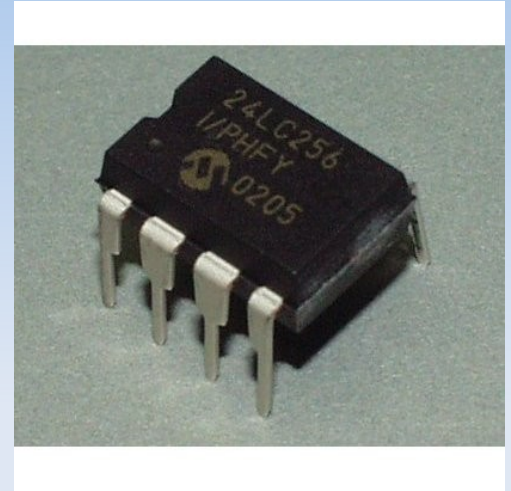
Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Typical 256 KBytes



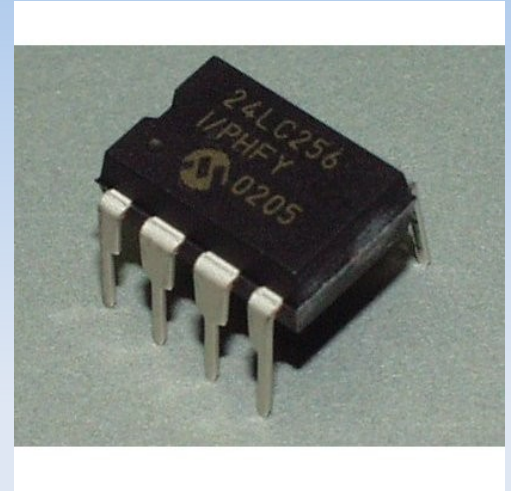
Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Typical 256 KBytes
 - Can be Write Protected



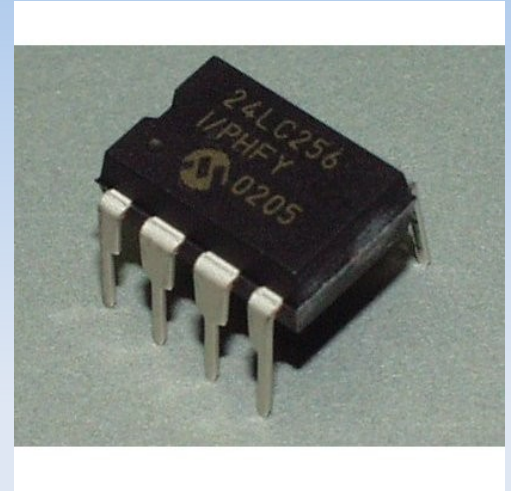
Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Typical 256 KBytes
 - Can be Write Protected
 - Low Cost



Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Typical 256 KBytes
 - Can be Write Protected
 - Low Cost
 - Multiples per System



Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Versioning

Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Versioning
 - EDID
 - Extended
 - Display
 - Identification
 - Data



Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Versioning
 - EDID
 - BeagleBone Capes



Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Versioning
 - EDID
 - BeagleBone Capes
 - Part/Board Identifications



Board Bringup

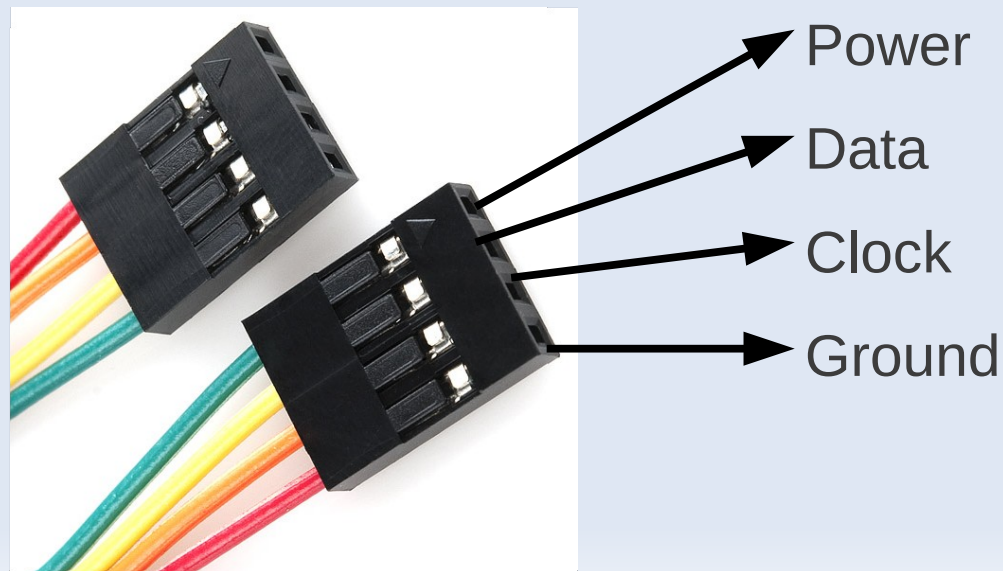
Name	Offset	Size (bytes)	Contents
Header	0	4	0xAA, 0x55, 0x33, 0xEE
EEPROM Format Revision	4	2	Revision number of the overall format of this EEPROM in ASCII = A0
Board Name	6	32	Name of board in ASCII
Version	38	4	Hardware version code for board in ASCII
Manufacturer	42	16	ASCII name of the manufacturer
Part Number	58	16	ASCII Characters for the part number
Number of Pins	74	2	Number of pins used by the daughter board
Serial Number	76	12	Serial number of the board. This is a 12 character string which is: WWYY4P13nnnn where: WW = 2 digit week of the year of production YY = 2 digit year of production nnnn = incrementing board number

Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Versioning
 - Debugging

Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Versioning
 - Debugging
 - Four Wire Connection



Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Versioning
 - Debugging
 - Four Wire Connection
 - Store Testing Cycle Data

Board Bringup

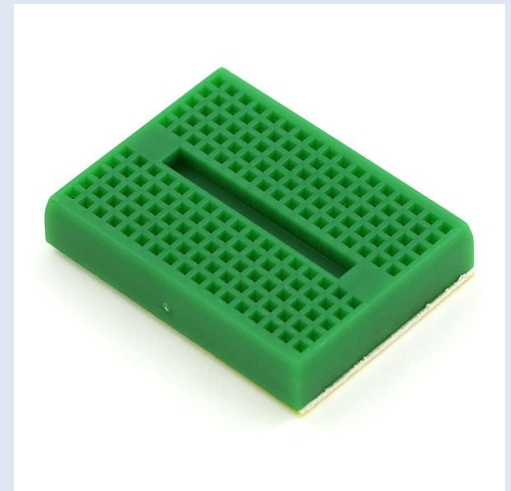
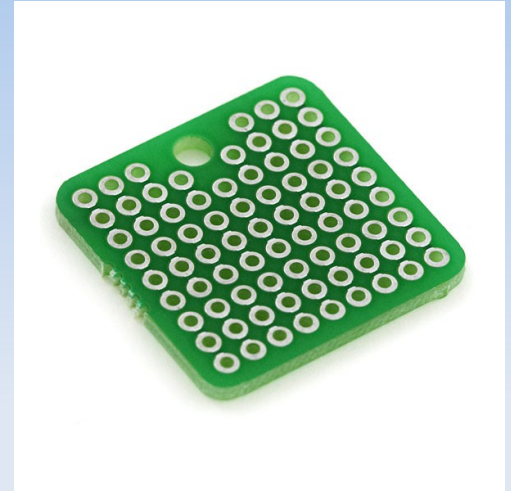
- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Versioning
 - Debugging
 - Four Wire Connection
 - Store Testing Cycle Data
 - Collect Board Interaction Data

Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Versioning
 - Debugging
 - Four Wire Connection
 - Store Testing Cycle Data
 - Collect Board Interaction Data
 - Configure Test/Boot Modes

Board Bringup

- I2C GPIO Expanders
- I2C EEPROMS
 - Devices
 - Versioning
 - Debugging
 - Four Wire Connection
 - Store Testing Cycle Data
 - Collect Board Interaction Data
 - Configure Test/Boot Modes
 - Easily Prototyped



Conclusion

- Communication Principles

Conclusion

- Communication Principles
- Drivers and Software Tools

Conclusion

- Communication Principles
- Drivers and Software Tools
- Board Bringup Use Cases

Conclusion



Questions?